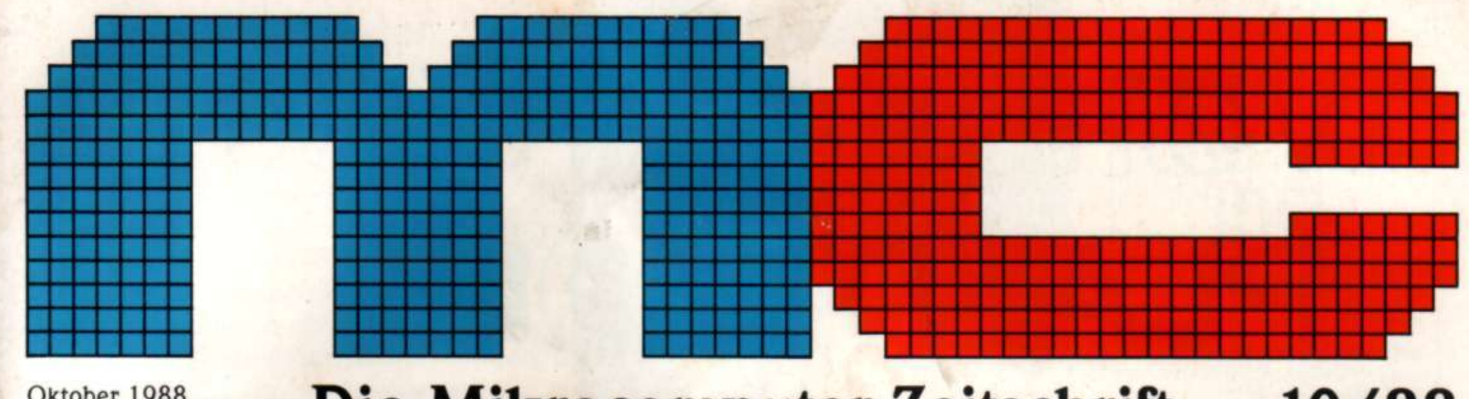


Oktober 1988
8 DM · 70 öS · 8 sFr

Die Mikrocomputer-Zeitschrift

10/88

Marktübersicht:

50 Scanner im Vergleich

Im Test:
Star-Parade
Kaypro-AT
Interquadram-Laptop
Siemens-Scanner
16-Bit-VGA

PC-Technik:
PC als Steuerungsrechner
So funktioniert die VGA
LAN im Detail

C-Workshop:
Stream-Editor

Macintosh:
DFÜ als Accessory

Modula:
SPC-Modula für Atari ST
Betriebssystem in Modula-2

EDITORIAL



Die Transputer-Aktivitäten in mc haben überraschend große Resonanz gefunden. Gedacht als Einstiegshilfe für alle die, die nach Einarbeitung in die Theorie praktische Erfahrung mit einem kleinen System sammeln wollen, hat sich die mc-Transputerplatine als Anstoß auch für uns erwiesen. In vielen Gesprächen mit Personen aus der

Liebe Leser!

Szene und solchen, die darin eindringen wollen, zeigte sich, daß Kommunikationsbedarf besteht. Die Fachleute haben viel Wissen, das an den Mann gebracht werden muß und die Anwender wollen viel wissen. Deshalb veranstaltet die mc-Redaktion zusammen mit der Firma Intellex ein Transputerseminar. Für 198 DM Unkostenbeitrag können Sie von hochkarätigen Experten (zum Beispiel N.Spicer – Fa. Microway, T. King – Fa. Perihelion, H. Oakley – Transputer Users Group, A. Cullouch – Fa. 3L, J. Hoffmann – Fa. Intellex) alles über Transputer in Vortrag und Diskussion erfahren. Es geht um die Markt-Aussichten, die Anwendungsfelder, die Technik und die Software. Das Seminar ist für einen Tag geplant. Veranstaltungsort wird Frankfurt sein. Termin: 23. November 1988. Kontakt: Dr. Jürgen Hofmann, Fa. Intellex Expertensysteme und Sensorik GmbH, Bettinastr. 14–16, 6000 Frankfurt am Main 1, Tel. (0 69) 75 20 23.

mc *kostet mit Ausgabe 10 im Handel 8 DM. Den Preis von mc konnten wir zwei Jahre stabil halten. Damit wir die redaktionelle Qualität halten und verbessern können, war dieser Schritt notwendig.*

In dieser Ausgabe von mc beginnt eine Serie über Arithmetik. Darin werden die Rechenweisen nach IEEE diskutiert. Das nicht nur in Bezug auf die Theorie. Es werden im Laufe der Serie Gleitkomma-Routinen für den 68 000, den 8086 und den Z80 vorgestellt. Vor allem interessant für den Einsatz in kleineren Systemen in Technik und Wissenschaft.

Ich habe in mc Bildschirmtext oft kritisch beleuchtet. Tatsächlich ist Btx als Medium in ganz anderer Art gewachsen als geplant. Das sei diesem Medium nicht weiter vorgehalten, denn es entwickelt sich mehr und mehr positiv. Der Franzis-Verlag tritt deshalb als Btx-Anbieter auf. Die Kollegen von der Zeitschrift Funkschau aus unserem Hause, der Zeitschrift für Telekommunikation, betreuen die Aktivitäten kraft Amtes federführend. Falls Sie mc eine Mitteilung über das Medium Btx machen wollen, hier die Nummer der Leitseite ∗ 3550# bzw. ∗ Franzis-Verlag#.

Es gibt eine Revolution in Bezug auf die Busarchitektur bei Industriestandard-Computern. Die Firmen Zenith, HP, Compaq, Olivetti, Tandy, Epson, NEC, AST, Intel, Microsoft, OCA, 3COM, Novell haben sich zusammengeschlossen und einen neuen Industriestandard für PCs angekündigt. Der Standard und die Gruppe der Firmen trägt den Namen EISA (von Extended Industrial Standard Architecture).

Das Anliegen der EISA ist es, den schätzungsweise 14 Millionen Anwendern von Personal Computern mit ISA (Industrial Standard Architecture, alias PC-Bus) einen Standard für die neue 32-Bit-Prozessorgeneration anzubieten. EISA ist eine aufwärtskompatible Weiterentwicklung des PC-Busses mit dem Ziel, mehr als 16 MByte Hauptspeicher (4000 MByte) zu unterstützen, schnellen Datentransfer zu gewährleisten (Burst Modus) und Mehrprozessorbetrieb (Arbitration) zu ermöglichen.

Mit EISA vereint gegen den Micro Channel? Viel Spaß!

Ihr

Ulrich Rohde

INHALT

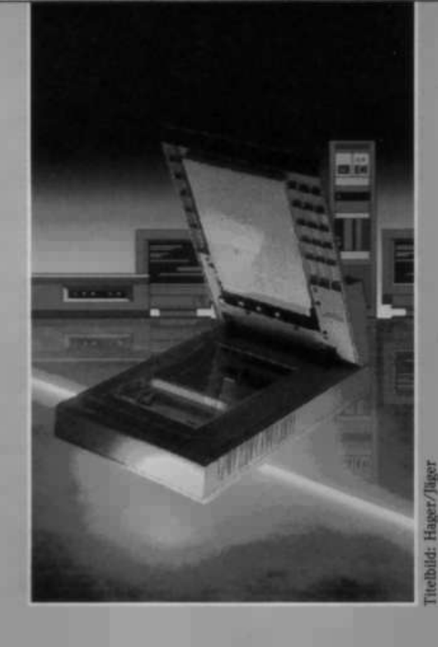
INFO

TEST

GRUNDLAGEN

Scan as scan can

Wer mit einem DTP-System professionell arbeiten will, kommt um einen Scanner nicht herum. Inzwischen gibt es vom Billig- bis zum Hochleistungs-Scanner für jeden Bedarf das richtige Gerät. Zu fast allen z. Zt. auf dem Markt befindlichen Scannern können Sie in unserer Marktübersicht die Daten vergleichen. **Seite 75**



Titelbild: Hager/Jäger

Daten im Netz

Netware von Novell für Fileserver-orientierte PC-Netze läßt sich in zwei Hauptkomponenten aufteilen: das Betriebssystem, das auf dem Fileserver läuft, und die Shell, die sich in den einzelnen Arbeitsplatzrechnern um das PC-Betriebssystem legt. Netware läuft derzeit auf insgesamt 43 Netzwerk-Hardware-Systemen, die meisten im Ethernet-, Token-Ring- oder Arcnet-Standard. Wichtige Eigenschaften von Netware haben wir unter die Lupe genommen. **Seite 70**



Modularer AT inklusive Software

Erst mit der Software wird ein Computer zu einem brauchbaren Handwerkszeug. Warum sollte ein PC-Hersteller nicht gleich einiges an Standard-Software beilegen, um damit den Kaufanreiz zu verstärken? Genau das hat Kaypro bei seinem AT K286 gemacht. Die Programme machen aus einem normalen 12-MHz-AT einen Renner. **Seite 54**



Laptop mit Anleitung zum Abspecken

Im Laptop-Markt tut sich einiges. Gefragt sind geringes Gewicht und Kompatibilität. Den Snap 1+1 von Interquadram kann man in zwei Teile zerlegen: einen Lapheld und ein Erweiterungs-Modul mit den Laufwerken. Und sogar die LCD-Anzeige läßt sich abnehmen. Damit ist der Snap 1+1 für vielfältige Anwendungen interessant. Eine mit 9,54 MHz getaktete NEC-V20-CPU verschafft ihm fast die Geschwindigkeit eines Rechners der AT-Klasse.

Seite 57

30 22 0
8-BIT EXP. | 23-BIT Mantisse | Einfache Genauigkeit
Mantissen-Vorzeichen
62 51 0
11-BIT EXP. | 52-BIT Mantisse | Doppelte Genauigkeit
Mantissen-Vorzeichen

Fließkomma-Arithmetik

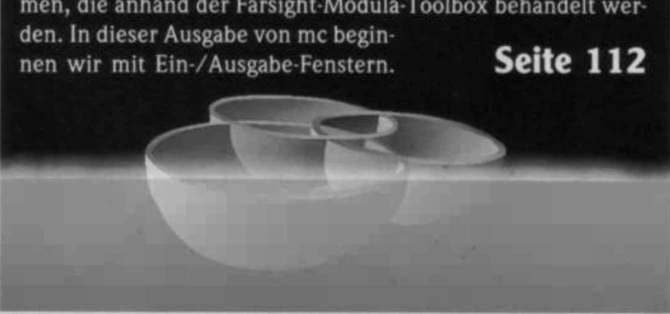
Um Computern das Rechnen beizubringen, bedarf es einiger Tricks. Die Numerik, ein Spezialgebiet der Mathematik, entwickelte dazu ihre eigenen Rechengesetze. Diverse Zahlenformate sollen insbesondere in punkto Genauigkeit jeder Anforderung gerecht werden. Standards und Strukturen sind das Thema der ersten Folge.

Seite 123

Ein Betriebssystem in Modula

Ein Blick hinter die Kulissen eröffnet eine neue Serie: sowohl für Leser, die selbst ein Betriebssystem schreiben wollen, als auch zum besseren Verständnis von Betriebssystemen. Parallele Prozesse und Programme, virtuelle Speicherverwaltung und die Definition des Betriebssystems sind einige der Themen, die anhand der Farsight-Modula-Toolbox behandelt werden. In dieser Ausgabe von mc beginnen wir mit Ein-/Ausgabe-Fenstern.

Seite 112



INHALT

BRIEFE

Parallele Prozesse in Modula-2

Zu dem Artikel von B. Götz und A. Meier (mc 6/88, S. 88) habe ich zwei Fragen:

1. DOS ist, wie bei Interruptgesteuerten Prozessen unangenehm auffällt, nicht „re-entrant". Diese Klippe wird von den Autoren auch anschaulich angesprochen und als Ausweg die Benutzung des BIOS-Interrupts 10H benutzt. Stimmt das so generell bzw. darf dies verallgemeinert werden zu „Das BIOS ist wiederaufrufbar"? Gibt es eine vollständige Aufstellung der DOS-Funktionen und BIOS-Interrupts mit der Angabe reentrant / nicht reentrant?
2. Die im Beitrag dargestellte Programmierung einer mitlaufenden Uhr ist nach meiner Meinung ein Beispiel für das „Konzept des Interrupts" und nicht für einen Scheduler, der den Rechner nach einem Zeitscheibenverfahren oder prioritäts-gesteuert abwechselnd mehreren rechenwilligen, wartenden Prozessen zuteilt. Die Uhr gibt die Kontrolle selbständig vor dem Eintreffen des nächsten Interrupts an den einen unterbrochenen Prozeß zurück und beendet damit ordnungsgemäß die Interruptbearbeitung. Ein Scheduler, der z. B. einmal pro Sekunde durch den Zeitgeber (1CH) aufgerufen wird, entzieht dem gerade unterbrochenen Prozeß den Rechner und teilt ihn mit TRANSFER einem anderen Prozeß zu. Entstehen dabei nicht immer tiefere Aufruf-Verschachtelungen, weil IRET fehlt? *K.-P. Becker, 6800 Mannheim*

Antwort des Autors:

1. MS-DOS, wie es uns heute vorliegt, ist nicht reentrant. Dies darf ohne weiteres verallgemeinert werden. Der Grund liegt, wie Ihnen wahrscheinlich auch bekannt ist, darin, daß nicht mehr als eine einzige Interruptebene zur Verfügung steht und somit beim Wiederaufrufen einer DOS-Funktion nach einem Interrupt ein in der Regel veränderter, wenn nicht gar undefinierter Zustand vorliegt. Unseres Erachtens wäre es bis auf ein paar Knacknüsse kein großer Aufwand, DOS reentrant zu machen, Modula-2 würde sich hierfür direkt anbieten. Unseres Wissens gibt es keine Aufstellung, die bezeichnet, ob und welche DOS/BIOS-Funktionen reentrant sind, insbesondere weil hier jede DOS-Version von der andern abweichen kann, je nachdem wie die einzelnen Funktionen programmiert wurden. Mit dem Wissen um nur eine Interruptebene gibt es aber eine sehr einfache Faustregel, welche Funktionen nicht für nebenläufige Prozesse verwendet werden sollen; solche nämlich sind gefährlich, welche lange dauern, d. h. besonders I/O-Funktionen jeder Art. Die Wahrscheinlichkeit ist bei diesen besonders hoch, unterbrochen zu werden. Damit ist auch schon die Antwort auf Ihre Frage, ob das BIOS generell wiederaufrufbar sei, angedeutet. BIOS-Funktionen sind in der Regel sehr kurz. Damit ist nicht nur die Wahrscheinlichkeit gering, daß diese unterbrochen werden, sondern die „heiklen" maskieren während der Ausführung den Interrupt, so daß sie gar nicht unterbrochen werden können und hiermit keine Wiederaufrufbarkeit vorgesehen werden muß. Wiederum läßt sich die Frage also nicht pauschal beantworten, da die Implementation von Fall zu Fall unterschiedlich aussehen kann. Sicherheit gibt nur der Blick in die Programmierung der jeweiligen Funktion.

2. Ihre Feststellung, daß der Beitrag eher ein „Konzept des Interrupts" darstellt als die Funktionsweise des Schedulers, freut uns sehr, da es uns offensichtlich gelungen ist, zu zeigen, wie einfach in Modula-2 Scheduler zu implementieren sind; so einfach nämlich, daß sie sogar fast übersehen werden. Im Beispiel wird nämlich wirklich ein Scheduler vorgestellt, der allerdings bloß einen einzigen Prozeß steuert. Wenn Sie nun jedoch den Scheduler um nur ein weiteres Transfer erweitern, dann haben Sie schon zwei Teilnehmer. Sie finden im Beispiel sogar eine Prioritätssteuerung, indem nämlich eine Variable dekrementiert wird und den Prozess erst nach mehrmaligen Durchlauf (= erniedrigte Priorität) ausführen läßt. Zu diesem Thema gibt es übrigens eine sehr interessante Publikation von Prof. Wirth: Schemes for Multiprogramming and their Implementation in Modula-2. Außerdem haben wir noch eine ganze Serie von Artikeln zu diesem Thema in petto, die wir bei entsprechender Nachfrage natürlich gerne ausarbeiten und veröffentlichen.

Nun aber noch zu Ihrer Frage. Auf einem Rechner mit MS-DOS als Betriebssystem liegt die Verantwortung zur Verwaltung von Interrupts beim Programm, nicht beim Betriebssystem, da dieses dazu ja gar nicht in der Lage ist, sonst wäre es ein Multitasking-System. Dank der Möglichkeiten von Modula-2 und den Compilern, die diese auch unterstützen, tragen Sie als Modula-2-Programmierer nur einen ganz kleinen Teil der Verantwortung, alles übrige wird Ihnen vom Compiler, genauer vom Run Time System abgenommen. So können Sie also auch sicher sein, daß Ihr Modula-2 System keine „unendlichen" Aufruf-Verschachtelungen „vorprogrammiert", außer Sie würden dies ausdrücklich so wollen. Sie könnten z. B. innerhalb einer Interruptabarbeitung eigenhändig einen weiteren Interrupt auslösen, der zur Verschachtelung führt. In einem solchen Fall wären Sie dafür verantwortlich, daß Ihr Programm reentrant ist. Wenn Sie übrigens den letzten Gedanken verstanden haben, besitzen Sie den Schlüssel zur sicheren Antwort auf Ihre erste Frage.

Albert Meier, CH-8906 Bonstetten/ZH

Wer kennt Apple-II?

Ich bitte Sie um Unterstützung bei einem technischen Problem: Für meinen technisch überholten, aber für den Hausgebrauch völlig ausreichenden Apple II+ – Kompatiblen habe ich mir ein FD – Laufwerk TEAC 55 FR besorgt. Mit diesem Laufwerk lassen sich keine Disketten formatieren, lesen oder beschreiben. Der Antriebsmotor und der Schreiblesekopf werden jedoch offensichtlich gesteuert.
Die FD-Controllerkarte ist die FDC4 von Ehring; ein Laufwerk TEAC 55F (mit zusätzlich eingebautem Kondensator) wird reibungslos bedient.
Wissen Sie eine Lösung des Problems? Welche HW-Modifikationen müssen vorgenommen werden? Kann dieses LW auch (durch einen weiteren HW-Eingriff) auf 40 Tracks gestellt werden? *Michael Wonczak, 8130 Starnberg*

Erfahrung mit „Kompatiblen"

In mc 12/86 war ein kurzes Assemblerprogramm abgedruckt, mit dem man ein Menü aufrufen kann, wenn die Tasten Shift-PrtScr, respektive Druck, betätigt wurden. Dieses Programm wurde von mir noch etwas ausgeweitet, aber die wesentlichen Funktionen sind gleich geblieben (ich habe jetzt 20 Menüpunkte).
Probleme gab es bis vor kurzem auf keinem PC, weder auf original IBM PC/XT, AT2 und AT3 noch auf einer Reihe von kompatiblen Taiwan-Geräten. Vor einigen Wochen wurde ich von einem Freund um einen Rat gebeten, welchen Rechner er sich kaufen sollte, und ich nannte einige Geräte. Durch eine Firmenspende bekam er allerdings einen VICTOR VPC III und einen NEC-P7 Drucker. Diese Kombination sollte sich als sehr schlecht erweisen.
Die allgemeinen Installierungen wurden bereits vom Händler vorgenommen, das Gerät kam mit MS-DOS 3.20 (nicht PC-DOS!), und lief sehr ordentlich. Als es nun um persönliche Ausstattung ging, wurde die Sache kritisch.
Als erstes installierte ich HGC mit dem Befehl „HGC SAVE", damit sich der Bildschirm nach 5 Minuten abschaltet. Das tut

er auch – aber vollständig! Erst ein Warmstart bewegt das Gerät wieder zu „normalem" Erwachen, oder blindes Herausstasten aus dem Programm und Aufruf irgendeines DOS-Befehles.

Als zweites wollte ich das oben genannte Menüprogramm für den NEC-P7 installieren. Ruft man das Menü mit Shift-PrtScr auf, so wird der ganze Rechner blockiert und stürzt voll ab. Hier hilft nur noch ein Kaltstart, um das Gerät wieder zu beleben!

Vorsichtshalber habe ich mich erst gar nicht an Sidekick gewagt, sondern erst einmal VICTOR in Frankfurt konsultiert. Des Rätsels Lösung ist eine vollständige Eigenentwicklung des BIOS durch VICTOR, eigentlich Kyocera (diese Firma steht nämlich hinter VICTOR). Dieses BIOS ist nicht mehr kompatibel zum IBM-Standard, sondern weist eigene Entwicklungen auf, die vollständige andere Wege für BIOS-Funktionen beschreiben.

Kommentar von VICTOR zu meinen Erfahrungen: Alles, was sehr hardwarenah programmiert ist (HGC.COM. MENU.COM, etc.), wird auf diesem Rechner nicht 100prozentig funktionieren. Daß man damit aber den Anwenderkreis von PCs um einige nützliche Prográmmchen bringt, das haben die Leute bei VICTOR nicht bedacht.

Mein Freund wird wohl jetzt zähneknirschend auf Programme aus dem TSR-Bereich verzichten müssen, der Rest an kommerziellen Programmen läuft leider (oder besser zum Glück) ohne irgendwelche Probleme.

Zum NEC-P7 bleibt nur zu sagen, daß ich das gleiche Gerät ebenfalls besitze, aber das Problem nicht erlebt habe. Aus nicht erklärlichen Gründen wird er durch den Rechner aktiviert, obwohl keinerlei Tastendrücke erfolgt sind. Da ein Einzelblatteinzug moniert ist, wird eine Seite eingezogen, oben links die Ziffer 2, ASCII 253, gedruckt und das Papier ausgeschoben. Weder ein Einfluß von Hochfrequenz noch von Netzschwankungen ließen sich erkennen, selbst mein eigener NEC-P7 zeigt dieses Problem am VPC-III.

Hier noch eine Liste meiner Hardware:

- Original IBM AT-02, 6 MHz, 20-MB-Festplatte, 1,2-MB- und 360-KB-Diskettenlaufwerke, Math. Coprozessor;
- Original Hercules Grafikkarte, SW-Monitor;
- NEC-P7 Drucker.

Ich hoffe, Ihnen einen kleinen Erfahrungshinweis gegeben zu haben. *Rainer A. Hoffmann, 6370 Oberursel*

Videotext

Ich habe inzwischen den Videotext-Decoder aus mc Heft 3...4/88 mit einem Bausatz der Firma Elektronikladen Detmold aufgebaut. Der Aufbau gelang mir auch ganz gut. Nur bei den Transistoren T9 und T11 hatte ich Schwierigkeiten bei der Bestückung, ich hatte die Pins vertauscht. Jetzt läuft das Gerät aber prima.

Sehr schnell habe ich entdeckt, daß ich das Gerät auch sehr gut für mein Hobby „Videofilmen" gebrauchen kann. Durch die Möglichkeit, Texte vom PC in die Seitenspeicher des Decoders zu schreiben, kann man den Videodecoder gut als Titelgerät gebrauchen. Es stehen 3 Textgrößen und eine Menge von Blockgrafik-Zeichen zur Verfügung.

Besonders interessant ist der Mix-Modus. Hiermit lassen sich die Titel in ein Kamerasignal einblenden. Dadurch läßt sich auch der Nachteil des Gerätes, nur ein Monochrombild zu liefern, umgehen. Man kann mit der Kamera einen farbigen Hintergrund sowie auch Dias, Bilder usw. einblenden. Die Einblendung des Textes in ein Videorecordersignal ist mir nicht gelungen, da macht anscheinend die Synchronisation nicht mit, obwohl das Datenblatt des Bausteins SAA5240 diese Möglichkeit beschreibt.

Eine Variante des Mix-Modus ist der Box-Modus. Hierbei wird der Text innerhalb einer schwarzen Box, deren Größe man selbst bestimmen kann, in das Bild eingeblendet. Diese ganzen Möglichkeiten kann man natürlich nur ausschöpfen, wenn man die Datenblätter der VALVO-Bausteine zur Verfügung hat. Ich habe mir diese bei der Firma VALVO in Hamburg besorgt.

Wenn es jetzt noch eine Möglichkeit gäbe, den Text in Farbe als FBAS-Signal zu liefern, wäre das Gerät wirklich vollkommen.

Gerhard Görlach,
6507 Ingelheim

Wir in den USA ...

Hätte Ray Duncan einen Blick in die Dokumentation zum UNIX-Assembler as geworfen, wäre ihm sicher nicht entgangen, daß mit dem Switch -m der Makroprozessor m4 aufgerufen wird, der sich aufgrund seiner externen Position durch eine wesentlich größere Flexibilität auszeichnet als dies ein bloßer Makroassembler könnte. In diesem Zusammenhang sollte man auch nicht übersehen, daß ein gewisses Betriebssystem für segmentierte Alpträume ganz ohne Assembler und mit einem Witz von „Handbuch" zur Auslieferung gelangt.

Als barock (=„schiefrund", verschnörkelt, übertrieben) würde ich die UNIX-Benutzeroberfläche nun gerade nicht bezeichnen, eher als von spartanischer Klarheit und damit genau richtig für Leute, die es stört, von ihrem Computer zum nagetierschiebenden Ikonenklicker degradiert zu werden.

Edmund Ramm,
2358 Kaltenkirchen

CodeView und Turbo-C

Nun habe ich es schon oft gehört und gelesen, Turbo C und CodeView passen nicht zusammen. Es werden sogar schon Programme wie TC Debugg angeboten. Aber es geht doch!

Der Haken liegt bei den Linkern. TLINK ist nicht in der Lage, eine für CodeView aufbereitete EXE-Datei zu erzeugen. Man erhält trotz eingeschalteter Line-numbers (in der TC-Option Compiler -> Code generation) die Meldung „No symbolic information". Wird zum Binden jedoch LINK von Microsoft benutzt, ändert sich das Bild schlagartig. Ein Aufruf von LINK sieht z. B. so aus:

```
c:link /co turboc start c0s test-
       ,test,, turboc lib cs
```

Der Schalter /co weist LINK an, die EXE-Datei für CodeView mit den benötigten Informationen zu versorgen. Der C-Quelltext muß jedoch gesondert bereitgestellt werden.

Ein Aufruf von CodeView mit test läßt jetzt das gewünschte Bild erscheinen.

Die von mir benutzten Programme haben folgende Versionen:

Turbo C 1.5
CodeView 2.00
Link 3.60

Roland Borris
4773 Möhnesee

Leserbrief von Herrn R. Scharf

Die Kritik, die Herr Scharf am Demo-Programm geübt hat, ist sicherlich berechtigt und sein Vorschlag sollte aufgegriffen werden. Allerdings ist es eine Auffassungsfrage, ob bei einem Programm, das gerade 1½ Druckseiten lang ist, eine derartige Fehlerabfrage durchgeführt werden soll oder nicht. Führt nämlich ein Aufruf zu einem Fehler, stimmt im System sowieso nicht mehr alles und für den Fall, daß auf einer Maschine der Speicherplatz nicht mehr ausreicht, scheitert bereits der Aufruf OpenScreen().

„Lebensnotwendig" wird eine korrekte Fehlerabfrage meiner Meinung nach erst dann, wenn man mit selbst-eingerichteten Ports arbeitet. Hier wird die geringste Unterlassung mit einem katastrophalen Absturz quittiert.

Christian Neubauer,
Nürnberg

C&C
Computers and Communications

NEU

NEC Pinwriter P6/P7 plus.
Mehr Schriften, mehr Speicher, mehr Möglichkeiten.



Die neuen NEC Pinwriter P6/P7 plus sind das Ergebnis konsequenter Weiterentwicklung der bewährten und zuverlässigen Bestseller P6 und P7: Einfache Bedienung, viele Schriftarten und enorm große Speicher. Beide Drucker sind blitzschnell (bis zu 265 Zeichen/Sek.) und garantieren bei einer hohen Auflösung von 360 x 360 dpi feine Grafik- und Schriftdarstellungen, die den Namen „Letter Quality" wirklich verdienen.

NEC

Weitere Informationen erhalten Sie von: NEC Deutschland GmbH, Klausenburger Straße 4, 8000 München 80, Tel.: 0 89/9 30 06-0, Telefax: 0 89/93 77 76/8, Telex: 5 218 073 und 5 218 074 nec m

Dieter Strauß

VGA für schnelle PC

V-RAM-VGA von Video Seven

Auf dem Grafiksektor tut sich einiges. Fast täglich erscheinen neue VGA-Karten am Markt. Die getestete V-RAM-VGA von Video Seven ist die erste VGA, die mit Video-RAM-Bausteinen bestückt ist. In schnellen 80286- und 80386-Systemen entfaltet sie erst richtig ihre flotte Video-Ausgabe.

Die V-RAM-Karte (*Bild 1* a, *Bild 1* b) von Video Seven weicht in zwei Dingen wesentlich von den herkömmlichen VGA-Karten ab: Sie ist für einen 16-Bit-Steckplatz konzipiert und ihr Bildspeicher ist mit Video-RAM-Bausteinen bestückt. Es ist auch genügend Platz für weitere Video-RAM-Bausteine auf der Karte vorhanden: Der Bildspeicher ist bis auf 512 KByte erweiterbar. Wenn man bedenkt, daß der maximale Arbeitsspeicher unter MS-DOS nur 640 KByte groß sein darf, ist das ist ein stolzer Wert für den Bildspeicher. Mit 512 KByte RAM stellt die Karte bei 1024 × 768 Bildpunkten 16 Farben gleichzeitig dar.

In welchen Fällen braucht der Anwender einen so großen Bildspeicher? Zur Darstellung von Schattierungen und Farbverläufen muß eine Grafikkarte viele Farben gleichzeitig darstellen können. Eine solche Anforderung tritt in der Praxis in technischen Anwendungen auf. Wenn man z. B. mit Autocad ein Maschinenteil konstruiert und mit dem Zusatzprogramm Autoshade eine möglichst naturgetreue Wiedergabe des Teils am Bildschirm erreichen will, wird von der Grafikkarte gefordert, daß sie bei hoher Auflösung viele Farben gleichzeitig darstellt. Die V-RAM-VGA kann bei 512 KByte Video-RAM und einer Auflösung von 720 × 540 Bildpunkten 256 Farben gleichzeitig darstellen. Falls sie nur mit 256 KByte RAM bestückt ist, bringt sie bei einer Auflösung von 1024 × 768 Bildpunkten immerhin noch vier Farben gleichzeitig auf den Bildschirm. In beiden Fällen kann man die dargestellten Farben aus einer Palette von 262144 Farben auswählen. Ein Farbpaletten-Baustein vom Transputer-Hersteller Inmos macht's möglich.

Tabelle: Die erweiterten Video-Modi

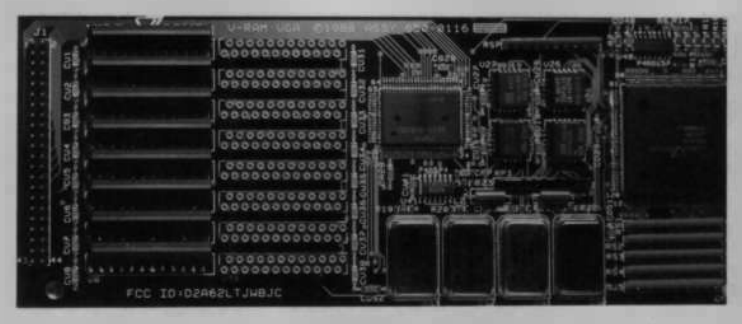
Video-Modus	Auflösung	Farben	Benötigter Monitor
40	80 × 43	16	VF, A
41	132 × 25	16	VF, A
42	132 × 43	16	VF, A
43	80 × 60	16	VF, A
44	100 × 60	16	VF, A
45	132 × 28	16	VF, A
60	752 × 410	16	VF, A
61	720 × 540	16	VF
62	800 × 600	16	VF
63	1024 × 768	2	HVF
64	1024 × 768	4	HVF
65	1024 × 768	16	HVF
66	640 × 400	256	VF, A
67	640 × 480	256	VF, A
68	720 × 540	256	VF
69	800 × 600	256	HVF

VF: Monitor mit variabler Synchronisation
HFV: Monitor mit einer sehr hohen variablen Synchronisation z. B. NEC XL)
A: Analog-Monitor (Festfrequenz)

Ein wesentliches Kriterium beim Test einer VGA-Karte ist ihre Kompatibilität zum Original von IBM. Gerüchte gehen um, daß es kompatible VGA-Karten gar nicht geben könne, da IBM nicht das komplette Innenleben ihrer VGA offengelegt habe. Was nicht bekannt ist, kann man schließlich auch nicht testen! Unterschiede bestehen sogar zwischen der VGA wie sie IBM in der PS/2-Serie verwirklicht hat und dem IBM PS/2 Display Adapter für den PC-/AT-Bus. In den PS/2-System erzeugt die Original-VGA am Ende des Bildrücklaufimpulses (horizontal blanking interval) einen Interrupt, der PS/2 Display Adapter jedoch nicht. Bei der V-RAM-VGA kann man deswegen mit einer Steckbrücke (JMP1) den Interrupt freigeben oder sperren.

Professionelle Grafik

mc hat zunächst einmal die V-RAM-VGA mit einigen professionellen Grafikprogrammen getestet, darunter Harvard Graphics, Autocad und der Windows-Applikation Micrografx Designer. Auch das DTP-Standard-Programm – der Pagemaker von Aldus – und zahlreiche VGA-Demo-Programme liefen ordnungsgemäß. Um den Test zu vervollständigen, habe ich die Karte mit dem VGA-Diagnose-Programm GRX von Renaissance traktiert. Mit diesem Programm werden einige selten gebrauchte Eigenschaften der VGA wie horizontales Scrollen, geteilter Bildschirm (Split Screen) und Row Mapping getestet. Ein Testprogramm von Chips & Technologies, VGATST genannt, schloß den Reigen der Prüfprogramme ab. VGATST stellt die Karte in den Modus mit einer Auflösung von 640 × 400 Punkten



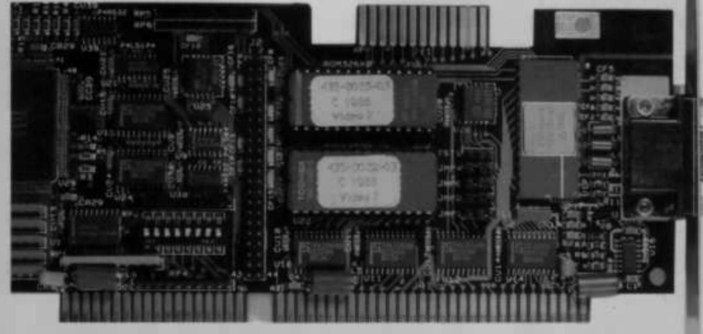


Bild 1a. Auf der V-RAM-VGA ist bereits Platz für weitere 256 KByte Video-RAM vorgesehen. Bild 1b. Fast alle Bausteine sind in SMD-Technik

ein und zeigt einen Ausschnitt aus einem mit Lotus 1-2-3 erstellten Kalkulationsblatt. Mit den Cursortasten kann man den Ausschnitt seitlich verschieben (Panning), was auch einwandfrei funktionierte. Allerdings fehlten in der unteren Bildschirmhälfte bei einigen Zeichen ein paar Bildpunkte. Der zweite Teil des Testprogramms schaltet die Karte in den Modus mit 1280 × 400 Bildpunkten um. Auf einem normalen Bildschirm werden die Zahlen allerdings so klein, daß man sie nur mit Mühe lesen kann.

Da ich noch kein kommerzielles Programm gesehen habe, das derart exotische Dinge mit der VGA macht, ist das Verhalten der V-RAM-VGA in dieser Betriebsart nicht von großer Bedeutung. Andere Video-Schnittstellen nach dem VGA-Standard, wie z. B. die im Testgerät (Mitac MPC2000VE) eingebaute VGA mit dem Paradise PVGA1-Chip hatten beim Panning wesentlich größere Probleme.

Bei einigen Rechnern, die allerdings nur wenig verbreitet sind, arbeitet die Karte erst nach dem Umstecken einiger Jumper einwandfrei. In solchen Fällen hilft die im Handbuch abgedruckte Fehlerbeseitigungsliste (frouble shooting list) weiter.

16-Bit-Bus

Ist eine 16-Bit-VGA schneller als eine 8-Bit-VGA? Die internen Register einer VGA sind 8 Bit breit. Was bringen da Wortzugriffe? Ich verglich daher die V-RAM-VGA mit der im Testgerät eingebauten 8-Bit-VGA und mit der VEGA VGA, die wie die V-RAM-VGA von Video Seven gefertigt wird.

In den Video-Modi 10H und 12H habe ich eine Bildschirmseite im Schreibmodus 0 hundertmal hintereinander beschreiben lassen und die Zeit gemessen. Das Testprogramm wird in der nächsten mc veröffentlicht und kann mit der nächsten MS-DOS-Sammeldiskette bezogen werden. Aus den in Bild 2 grafisch dargestellten Ergebnissen geht hervor, daß in der vorliegenden Testumgebung die V-RAM-VGA bis zu dreimal schneller als die anderen VGA-Karten ist. Man darf sich jedoch nicht zu der Aussage verleiten lasssen, daß die V-RAM-VGA generell dreimal schneller sei. Im Gegensatz zu einem Computer als einem gesamten System, hängt die Arbeitsgeschwindigkeit einer Systemkomponente stark von der jeweiligen Umgebung ab. Messungen mit dem von der amerikanischen Fachzeitschrift PC-Tech-Journal veröffentlichten Testprogramm ATPERF [1] haben ergeben, daß die V-RAM-VGA mit weniger Wartezyklen als andere VGA von der CPU bedient wird. Gemessen wird dies im Textmodus.

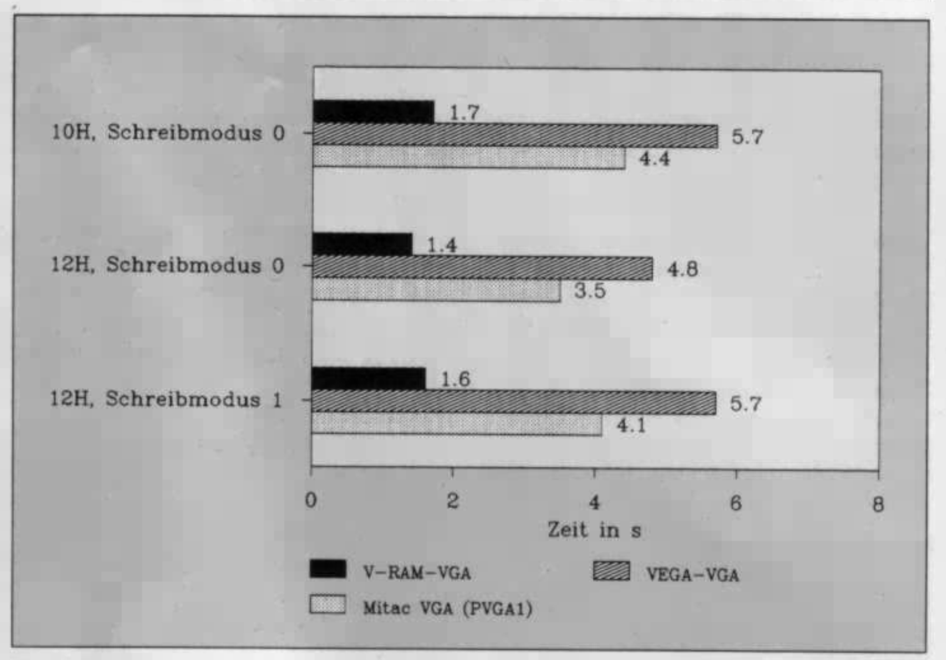


Bild 2. Im direkten Bildspeicherzugriff zeigt sich die Stärke der V-RAM-VGA

```
Die erweiterten Video-Modi werden mit folgenden Befehlen angesprochen:

Video-Modus einschalten:

mov ah, 6fh
mov al, 05h
mov bl, modus_nummer   ; Nummer aus Tabelle
int 10h

Abfrage: Ist V-RAM-VGA installiert?

mov ah, 6fh
mov al, 00h
int 10h

Wenn die V-RAM-VGA installiert ist, steht in
bh der ASCII-Code von "V" und in bl "7

Abfrage: Welcher Monitor ist installiert?

mov ah, 6fh
mov al, 01
int 10h

In al wird die Identifikations-Nummer des Monitors zurückgegeben.


Identifikations-        Monitor
  Nummer
    00h          unbekannter Monitor
    01h          Monochrom-Monitor
    02h          CGA-Monitor (200 Zeilen)
    03h          EGA-Monitor
    04h          Monitor mit variabler Sync. (NEC, NANAO)
    05h          Monitor mit variabler Sync. (Sony, JVC, Mitsubishi)
    10h          Festfrequenz-Monitor (IBM 8503, 8512, 8513, 8514)
```

Bild 3. Einige Assembler-Routinen für Programmierer

Die Register der VGA sind paarweise angeordnet, d. h. ein Adreß- oder Index-Register wird von einem Datenregister gefolgt. Mit einem 16-Bit-Ausgabe-Befehl werden bei einer 16-Bit-VGA beide Register quasi gleichzeitig programmiert. Bei einer 8-Bit-VGA muß die Bus-Logik des jeweiligen PC einen 16-Bit-Ausgabebefehl in zwei 8-Bit-Zugriffe zerlegen. Dies kostet Zeit.

Außer in Benchmark-Programmen macht die Karte auch in Anwenderprogrammen einen guten Eindruck. Vorausgesetzt eine schnelle CPU ist im System. Bei einem PC, der mit 4,77 MHz getakteten 8088-CPU läuft, ist eine 16-Bit-VGA unnötig. Ein schnelles 80286- oder 80386-System wird mit der V-RAM-VGA erst richtig zum Leben erweckt, was sich bei grafischen Benutzer-

oberflächen wie z. B. Windows oder Spielen bemerkbar macht.
Ein interessanter Aspekt der Karte sind die Video-RAMs selbst. Auf der Karte sind sie in der linken Ecke untergebracht. In Bild 1 sind deutlich die Fassungen für die Erweiterung auf 512 KByte VRAM zu sehen. Die VRAMs sind im ZIP-Gehäuse untergebracht. Ihre Speicherorganisation ist 64 KByte × 4. VRAMs werden manchmal als Dual Ported Dynamic Random Access Memory bezeichnet. Normale dynamische RAMs, die in Computersystemen als Arbeitsspeicher verwendet werden, können zu einem Zeitpunkt entweder nur gelesen oder beschrieben werden. In Video-Schnittstellen verwenden sie den gleichen Anschluß, um Daten von der CPU zu empfangen und über die Video-Logik an den Monitor weiterzuleiten. Die Daten müssen sich durch einen Kanal zwängen, d. h. der Datenfluß ist gehemmt. Bei VRAMs erfolgt der Zugriff auf Daten über zwei Kanäle.

Ein VRAM besteht, vereinfacht gesagt, aus einem dynamischen RAM und einem Schieberegister. Die CPU greift normal auf den Parallel-Ein-/Ausgang zu, während an dem anderen Anschluß die Daten bereits serialisiert herauskommen. Der Bildschirm-Refresh kann so bis zu zehnmal schneller als bei normalen Video-Speichern erfolgen, die mit dynamischen RAMs bestückt sind. Die hohe Geschwindigkeit der VRAM-VGA dürfte zu einem großen Teil auf die Verwendung der Video-RAMs zurückzuführen sein.
Wie schon die VEGA VGA [2] kann man auch die V-RAM-VGA in erweiterten Video-Modi betreiben. Die *Tabelle* gibt einen Überblick über diese erweiterten Video-Modi. Wenn sich die VRAM-VGA nur als reine VGA-Karte verhalten soll, muß man den DIP-Schalter 8 auf die Position On stellen.
Einige nützliche Codesequenzen für die V-RAM-VGA sind in *Bild 3* wiedergegeben.

Wer sich näher mit der Programmierung der VGA-Karten befassen will, dem empfehle ich das Buch Video Systems von Microsoft Press [3]. Außer auf die Programmierung von VGA, EGA, CGA, MCGA geht es auch die Karten von Hercules ein (inklusiv der Hercules Incolor Card).
Bei Grafik-Anwendungen mit hoher Auflösung auf schnellen 80286- und 80386-Systeme bringt die V-RAM-VGA einen deutlichen Geschwindigkeitszuwachs gegenüber herkömmlichen VGA. Derzeit kostet sie um 2104 DM und wird von Computer 2000 über Fachhändler vertrieben. Kundenfreundlich ist die fünfjährige Garantie.

Literatur

[1] *Forgeron, Pierce, Armbrust*: Updating the Evalution Suite. PC-Tech-Journal. März 1987, Seite 71.
[2] *Dieter Strauß:* Grafik mit der VEGA VGA. mc 1988, Ausgabe 3, Seite 58.
[3] *Richard Wilton:* Video Systems. Microsoft Press, Vieweg Verlag

Drei nützliche Programme für den Atari ST

Einer der Kritikpunkte am Atari ST ist immer wieder die Festplatte, die vielen Anwendern zu langsam erscheint. Aus dem überaus reichhaltigen Software-Angebot für den Atari ST stellen wir drei Utilities vor, die das Arbeiten mit der Festplatte angenehmer machen.
Zum Optimieren der Zugriffszeiten dient das Programm „Hard Disk Sentry" von Computerware Sender aus Köln. Es erkennt und behebt darüber hinaus Fehler auf der Festplatte.

Festplatten-Wächter

Durch die intensive Nutzung einer Festplatte und das ständige Löschen und Anlegen von Dateien wird die logische Struktur der Festplatte stark zerklüftet. Die Folge ist, daß viele Dateien über große Bereiche der Festplatte verstreut liegen und beim Einlesen in den Speicher erst zusammengesucht werden müssen. Optimal hingegen kann ein Computer mit der Festplatte arbeiten, wenn alle Blöcke einer Datei direkt hintereinander liegen. Für diese Aufgabe eignet sich „Hard Disk Sentry" hervorragend. Sentry heißt übrigens Wachtposten. Er stellt fest, welche Dateien nicht aufeinanderfolgende Blöcke belegen und organisiert die Festplatte so um, daß wieder alle Blöcke einer Datei aufeinanderfolgen. Bei unserem Test haben sich spürbare Beschleunigungen beim Starten von Programmen oder Einlesen von großen Dateien gezeigt. Nach dem Test der Festplatten-Partition zeigt das Programm alle wichtigen Informationen übersichtlich an. Besonders die Anzahl der unterbrochenen Blöcke ist ein Maß dafür, ob nach dem Test eine Optimierung durchgeführt werden sollte. Bei der Optimierung kann man angeben, wie groß der Bereich zwischen den Blöcken einer Festplatte sein soll und ob der nicht mehr benötigte Bereich für bereits gelöschte Unterverzeichnisse wieder freigegeben werden soll.
Sind bei der Arbeit mit einer Festplatte fehlerhafte Dateien entstanden, erkennt das Programm das und bietet die fehlerhaften Dateien zur „Reparatur" an. Alle Meldungen des Programmes und das Handbuch sind in deutscher Sprache und leicht verständlich. Die Ausgabe von 140 DM für dieses Programm lohnt sich sicher.
Mit einem weiteren Gesichtspunkt des Festplatteneinsatzes – dem Backup und Restore – setzt sich das „Harddisk Utility" von Application Systems aus Heidelberg auseinander. Die zum Betriebssystem MS-DOS gehörenden Befehle Backup und Restore sind im Betriebssystem des Atari ST nicht enthalten. Mit dem Programm HDU, das für 69 DM erhältlich ist, wird dieser Mißstand beispielhaft ausgeräumt. Nach dem Markieren der zu sichernden Dateien (über Datum, Extension, Attribut oder Auswahl mit der Maus-Taste) werden sie sehr schnell auf Disketten gesichert. Die Dateien können auf Wunsch in einem komprimierten Format auf den Sicherungs-Disketten abgelegt werden. Sollte die Ziel-Diskette noch nicht formatiert sein, bietet das Programm eine Formatierung während des Backup-Vorganges an.
Beim Restaurieren von Datenbeständen muß man sich entscheiden, ob bestehende gleichnamige Dateien überschrieben werden sollen, ob man wahlweise überschreiben möchte oder ob man generell gleichnamige Dateien nicht überschreiben möchte. Alles weitere übernimmt die Software, die ausgereift und einfach zu bedienen ist.

RAM-Disk-Utility

Als Alternative zur Festplatte bietet sich bei den Atari-ST-Rechnern eine RAM-Disk an. Eine RAM-Disk der Superklasse bietet Application Systems aus Heidelberg für 69 DM an. Das Programm „Flexdisk" ist unter GEM, TOS und als Accessory lauffähig. Die RAM-Disk ist auf Wunsch speicherresident und kann dynamisch dimensioniert werden. Dadurch belegt sie nur soviel Speicherplatz wie erforderlich. Beim Booten des Computers werden Dateien, die in einer Steuer-Datei aufgeführt sind, aus allen angeschlossenen Laufwerken auf die RAM-Disk kopiert. Zu diesem Programm gehört noch eine nützliche Kopier-Utility.
Alle drei Programme sind ihr Geld wert und sehr hilfreich. *rs*

Günther Sternberg

Hochauflösender Flachbett-Scanner

Der ST400 von Siemens arbeitet mit 64 Graustufen

Ein wichtiges Qualitätskriterium beim Kauf eines Scanners ist seine Auflösung. Unser Test hat es an den Tag gebracht, daß allein eine hohe Auflösung noch lange keinen guten Scanner ausmacht. Ein guter Scanner muß Graustufen verarbeiten können, und Schwellwerte müssen einstellbar sein.

Desktop Publishing, meist kurz DTP genannt, gilt derzeit als eines der innovativsten Gebiete der EDV mit einem rasch expandierenden Markt. Die Integration von Text und Bild in einem mit dem Computer bearbeitbaren Dokument ist ohne Scanner kaum denkbar. Kein Wunder, daß die Hersteller von Scannern ständig neue Geräte auf den Markt bringen. Aufgrund der durch den DTP-Boom ausgelösten Massenproduktion sind hochwertige Scanner bereits unter 10 000 DM erhältlich.

Der ST400 von Siemens (*Bild 1*) ist ein Schwarzweiß-Flachbett-Scanner für eine maximale Größe der Vorlage von DIN A4. Mit seiner maximalen Auflösung von 400 dpi (dots per inch) ragt der ST400 aus der Masse der Standard-Scanner hervor, die nur eine maximale Auflösung von 300 dpi haben. Seine Arbeitsweise ist einfach zu verstehen. Wie beim Fotokopierer liegt die Vorlage auf einer Glasplatte und wird von unten abgetastet. Im Inneren des Gerätes bewegt ein Schrittmotor die Abtastvorrichtung zeilenweise über die Vorlage. Auf ihr sind zwei Zeilen mit sehr hellen Leuchtdioden angebracht. Das von den Leuchtdioden abgestrahlte Licht wird von der Vorlage reflektiert und über Glasfaser-Lichtleiter auf CCD-Sensoren (charge coupled devices) gelenkt. Die Sensoren messen die Intensität des zurückgestrahlten Lichts.



Bild 1. Ein Scanner mit kompakter und platzsparender Bauform

Eine weiße Stelle der Vorlage reflektiert das eingestrahlte Licht nahezu vollständig, während eine schwarze Stelle das Licht absorbiert. Die nachfolgende Elektronik setzt die Information „Licht" oder „kein Licht" in die für den Computer verständlichen Binärwerte „0" oder „1" um. Wie im täglichen Leben kommt man mit einer reinen Schwarzweiß-Betrachtungsweise nicht weit. Ein Schwarzweiß-Foto besteht nun mal nicht nur aus absolut schwarzen und völlig weißen Stellen, sondern aus vielen unterschiedlichen Grauwerten. Für eine originalgetreue Wiedergabe reicht daher eine reine Ja-Nein-Information nicht aus. Der ST400 unterscheidet zwischen 64 verschiedenen Grauwerten. Für jeden Pixel wird die Intensität des zurückgestrahlten Lichts gemessen, digitalisiert und als 6-Bit-Zahl abgelegt.

Die Ergebnisse können sich sehen lassen

Was eignet sich besser zum Testen eines Scanners als eine Vorlage mit einem möglichst feinen Muster. Jeder Leser hat so eine Vorlage beinahe täglich in der Hand – ich spreche von Geldscheinen. *Bild 2* zeigt den Ausschnitt eines mit 400 dpi, ohne Verwendung von Grauwerten, gescannten 10-DM-Geldscheins. Trotz der hohen Auflösung sieht der Geldschein unnatürlich aus. Gerade bei Geldscheinen werden zur Erhöhung der Sicherheit gegen Fälschungen viele unterschiedliche Farbschattierungen verwendet. Sie werden erst bei der Verwendung von Graustufen (*Bild 3*) berücksichtigt. Da ein Laserdrucker Grauwerte nur indirekt durch Variation der Farbdichte pro Flächeneinheit darstellen kann, der Toner hat nun mal nur eine feste Schwärzung, geht die Nutzung der Grauwerte leider zu Lasten der effektiven Auflösung. Trotzdem sieht der Geldschein in Bild 3 wesentlich natürlicher aus als der mit der besseren Auflösung gescannte Schein in Bild 2.

Siemens stellte der Redaktion außer dem Scanner, der SCSI-Einsteckkarte und dem zugehörigen Verbindungskabel auch eine 2-MByte-Speichererweiterungskarte von Tall Tree Systems zur Verfügung. Dies hat einen guten Grund, wie sich beim Studium des Scanner-Handbuchs zeigte. Das 108seitige deutsche Handbuch enthält leider nur Installationsanweisungen für die Speichererweiterung von Tall Tree Systems und dem Intel Above Board. Wer eine Speichererweiterung eines anderen Herstellers besitzt, (die Karte muß dem EMS-Standard entsprechen) steht vor einigen Problemen. Es lassen sich auch andere Speichererweiterungen verwenden. So konnte ich mit einer Noname-AT-RAM-Bank erfolgreich arbeiten. Ohne Speichererweiterung geht wegen der riesigen Datenmengen überhaupt nichts. Z. B. belegt eine mit 64 Grau-

stufen gescannte DIN A4-Seite bei 400 dpi ungefähr 15 MByte Speicher. Dementsprechend sollten auf der Festplatte noch mindestens 17 MByte frei sein.

Außer einem 16-Bit-Steckplatz für die Speichererweiterung wird noch ein kurzer 8-Bit-Steckplatz für das SCSI-Interface benötigt. Diese Hochgeschwindigkeits-Schnittstelle überträgt 32 MBit pro Sekunde. Das Scannen einer DIN A4-Seite mit 64 Graustufen und 300 dpi benötigt inklusive Datenübertragung zum Computer (10 MHz-AT) etwas über 5 Minuten. Die Seite wird dabei übrigens in sechs Teilen gescannt und in Mengen von je 2 MByte zum Host-Rechner übertragen. Der Einbau und die Konfiguration der Karten, sowie die Inbetriebnahme des Scanners anhand des Handbuchs klappten auf Anhieb einwandfrei. Benutzt man die von Siemens empfohlenen Karten, ist das mit vielen Bildern illustrierte Handbuch eine große Hilfe.

Für Anwender eines Laser-Druckers mit einem Canon-Druckwerk, zum Beispiel einem HP Laserjet II oder dem QMS PS-800, gibt es noch einen anderen Grund, die Speichererweiterung JRAM AT3 von Tall Tree Systems zu benutzen. Auf der Speichererweiterungskarte kann man eine Huckepack-Karte mit der Bezeichnung JLaser installieren. Bei dieser Zusatzkarte handelt es sich um eine Video-Schnittstelle zum Aufbau einer schnellen Übertragungsleitung zum Laser-Drucker. Der Laser-Drucker wird dabei mit dem entsprechenden Gegenstück, der JSX-Karte, aufgerüstet. Diese Aufrüstung ist ratsam, da die Ausgabe einer DIN-A4-Seite selbst über die schnellere parallele Schnittstelle (Centronics) wegen der niedrigen Übertragungsrate sonst leicht fünf Minuten dauern kann. Leider läßt sich der verwendete Kyocera-Laser-Drucker (F-1010) nicht mit dieser Karte aufrüsten, so daß eine praktische Erprobung nicht möglich war.



Bild 2. Ohne Verwendung von Grauwerten: Nehmen Sie mal einen 10-DM-Schein zur Hand und vergleichen Sie selbst



Bild 3. Mit Verwendung von Grauwerten: Trotz geringerer Detailtreue scheint das Bild natürlicher zu sein



Bild 4. Der Bildausschnitt läßt sich mit den für Windows typischen Schiebern verändern

Hilfsprogramme unter Windows 1.03

Der Nutzen eines Scanners hängt in erster Line von der mitgelieferten Software für den PC ab. Die vom Scanner angelieferte Datenflut muß kanalisiert und je nach Anwendungszweck aufbereitet werden. Die möglichst unkomplizierte Einstellung wichtiger Scan-Parameter, wie Auflösung, Schwellwerte und Vorlagengröße, ist eine weitere Forderung an die Software.

Die Software des ST400 setzt einen IBM-AT-kompatiblen Rechner mit 640 KByte Hauptspeicher und mindestens 2 MByte Expanded Memory voraus. Das Programm läuft unter der Benutzeroberfläche Windows von Microsoft. Das Handbuch empfiehlt die Version 1.03 von Windows. Bei neueren Versionen von Windows soll es Probleme geben, die allerdings nicht näher spezifiziert sind. Leider liefert Siemens die passende Version von Windows nicht mit. Da Windows Version 1.03 nicht mehr im Handel erhältlich ist und die Software auf diese Version angewiesen ist, dürfte für viele Anwender die mitgelieferte Software nicht verwendbar sein.

Abgesehen davon ist die Software zum ST400 gelungen und sehr leistungsstark. So bekommt der Anwender eine gescannte Vorlage direkt am Bildschirm angezeigt (*Bild 4*) und hat 16 Stufen zum Vergrößern und Verkleinern des Bildes bei der Ausgabe am Bildschirm zur Verfügung. Die Einstellung der Scan-Parameter erfolgt über Dialogfenster und ist kinderleicht. Verfügbare Auflösungen sind 200, 300 und 400 dpi. Vorlagengrößen sind DIN A6, DIN A5 (hoch und quer) sowie DIN A4. Mit der Maus kann man einen beliebigen rechteckigen Ausschnitt festlegen. Ein gescannter Ausschnitt läßt sich verschieben und getrennt speichern.

Die Bildinformationen kann man in verschiedenen Dateiformaten speichern. So hat man die Wahl zwischen dem TIFF-Format (Tag Image File Format), das vom Pagemaker benutzt wird, oder dem GEM-Format, beispielsweise für den Ventura Publisher. Die Speicherung von Schwarzweiß-


Bild 5. Das Originalfoto als Maßstab für die Scan-Qualität

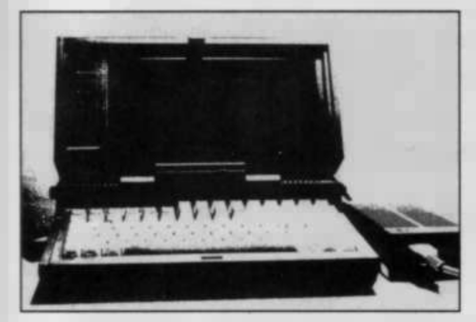

Bild 6. Details kann man bei dieser Schwarzweiß-Darstellung nur ahnen


Bild 7. Der Laptop mit 64 Graustufen und den Standard-Einstellungen der Software gescannt


Bild 8. Ein Anheben des Schwellwerts ist nur in geringem Umfang nützlich


Bild 9. Die softwaremäßige Nachbearbeitung des Bildes führt zu verblüffenden Ergebnissen

Die technischen Daten des ST400 auf einen Blick

Hardware:	
Bezeichnung	ST400
Hersteller	Siemens AG
Typ	Schwarzweiß-Flachbett-Scanner
Scanfläche	DIN A6, A5, A4 und beliebiger Ausschnitt
Auflösung	200, 300 und 400 dpi
Graustufen	64
Sensoren	CCD
Optik	Glasfaser-Lichtleiter
Speicher	2 MByte
Schnittstelle	
– Typ	SCSI-Interface (kurze 8-Bit-Steckkarte)
– Transferrate	32 MBit/s
Arbeitsgeschwindigkeit	DIN-A4-Seite mit 64 Graustufen und 300 dpi in 310 s DIN-A4-Seite ohne Graustufen mit 300 dpi in 24 s (benutzter Computer: Sanyo MBC 17 plus mit 8 MHz)
Software:	
Voraussetzungen	
– Computer	IBM-AT-kompatibler Rechner
– Betriebssystem	MS-DOS ab Version 3.1
– Benutzeroberfläche	MS-Windows Version 1.03
– Speicher	640 KByte Hauptspeicher, 2 MByte Speichererweiterung
Dateiformate	TIFF (z. B. für PageMaker) GEM (z. B. für Ventura Publisher)
Funktionen	Bildschirmanzeige der gescannten Vorlage, Vergrößerung und Verkleinerung der Bildschirmanzeige, Veränderung des Schwellwerts, Veränderung der Auflösung, Rasterung der gesamten Vorlagen (6 Raster), Festlegung von Ausschnitten, Speichern von Ausschnitten

Darstellungen und gerasterten Bildern ist bei beiden Formaten möglich, während Graustufen nur vom TIFF-Format unterstützt werden.

Zum Schluß sollen die Möglichkeiten der Software zum Aufbereiten eines gescannten Fotos vorgeführt werden. Dazu wurde das Foto (*Bild 5*) eines von der mc getesteten Laptops mit verschiedenen Parametern gescannt. Ohne Verwendung von Graustufen, also bei reiner Schwarzweiß-Darstellung, ergibt sich die in *Bild 6* gezeigte Qualität. Man kann zwar erkennen, daß es sich um einen tragbaren Computer handelt, die Unterscheidung von Einzelheiten ist nicht möglich. Allein die 64 Graustufen mit den Standardeinstellungen der Software bringen einen erheblichen Qualitätsgewinn (*Bild 7*). Jetzt lassen sich bereits Details so gut erkennen, daß man auf dem Original-Ausdruck den Namen des Testgeräts lesen kann. Störend bei diesem Bild ist der dunkle Hintergrund. Durch ein Anheben des Schwellwerts, also der „Empfindlichkeit" des Scanners (*Bild 8*) wird nicht nur der Hintergrund heller, sondern das ganze Bild. Damit verschlechtert sich der Kontrast. Das Bild wirkt jetzt wie eine überbelichtete Fotographie. Mit dem Hochsetzen des Schwellwerts allein, und bei einigen Scannern ist dies das einzige Hilfsmittel, kommt man zu keiner besseren Qualität. Die Lösung liegt nicht in der Wahl der geeigneten Parameter bei der Bilderfassung, sondern in einer Nachbehandlung des gescannten Bildes.

Mit der Software zum ST400 läßt sich der Kontrast nachträglich ändern. Die Grauwerte eines Bildes werden in die drei Kategorien unterteilt. Der Helligkeitsgrad jeder Kategorie läßt sich nachträglich durch einen geeigneten Algorithmus variieren. Beim *Bild 9*, das mit den gleichen Einstellungen wie Bild 7 aufgenommen wurde, ist die Helligkeit der hellen Zonen angehoben und somit der Hintergrund weitgehend aufgehellt worden.

Ein zukunftsorientiertes Gerät

Der ST400 wurde während des vierwöchigen Tests intensiv genutzt und erwies sich in dieser Zeit als modernes und zuverlässiges Hilfsmittel zum Erfassen vor Vorlagen aller Art. Mit seiner Auflösung von 400 dpi und den 64 Graustufen unterstützt das Gerät den Trend zu anspruchsvolleren DTP-Anwendungen. Der kompakte Scanner, der ca. 10 000 DM kostet, hinterläßt nicht zuletzt wegen gutdurchdachter Kleinigkeiten, wie z. B. der versenkten Vorderkante zum problemlosen Wechsel der Vorlage, einen guten Eindruck. Daran können auch die noch verbesserungswürdigen Details, wie die bei der Bewegung des Abtastbalkens wohl wegen einer Resonanz entstehenden Geräusche oder das in manchen Punkten doch sehr knappe Handbuch, nichts ändern. □

Günther Sternberg

Software im Preis inbegriffen

Ein modularer 12-MHz-AT: Kaypro K286

Allein Geschwindigkeit und moderne Technik reichen oft nicht aus, um einen Computer der AT-Klasse zu einem Erfolgsmodell zu machen. Mit umfangreicher Software bietet Kaypro einen zusätzlichen Kaufanreiz. Wir haben den Kaypro K286 gründlich unter die Lupe genommen.

Die Anschaffung eines Computers weist immer mehr Parallelen zum Kauf eines Autos auf. Man achtet zwar auf technische Daten wie Taktfrequenz, Speichergröße und Kapazität der Laufwerke und bestaunt anhand eines imposanten Demo-Programms die Leistungsfähigkeit der Rechner. Letztendlich sind es oft Kleinigkeiten, so beispielsweise ein ansprechendes Design, die bei annähernd gleichwertigen Geräten den Ausschlag zum Kauf eines bestimmten Computers geben.

Der K286, ein 12-MHz-AT des amerikanischen Herstellers Kaypro, ist ein typischer PC der AT-Klasse. Seine wichtigsten technischen Daten lauten:

- 80286 CPU
- 6/12 MHz Taktfrequenz (umschaltbar)
- 1 MByte RAM
- 40-MByte-Festplatte
- modularer Aufbau (keine Systemplatine)
- 9 Steckplätze
- EGA-Karte

Diese Daten sind, wie es so schön heißt, heute Stand der Technik und kennzeichnen beileibe keinen außergewöhnlichen PC mehr. Wie beim Autokauf sind es die im Preis inbegriffenen Extras, die ihn interessant machen. Die Liste der kostenlosen Zugaben beim Kaypro K286 ist eindrucksvoll. Außer den englischen Handbüchern zu Computer, MS-DOS und GW-Basic liefert Kaypro folgendes mit:

- zwei deutschsprachige Bücher von Markt & Technik mit den Titeln „PC-DOS/MS-DOS 3.2“ und „Effektives Programmieren in GW-Basic“ inklusive Disketten mit Übungen und Beispielen
- das Hilfsprogramm SpeedStor zum Formatieren, Partionieren und Verwalten von Festplatten samt zugehörigem englischem Handbuch
- Wordstar Version 4.0 mit Handbuch (englisch)
- Word Finder mit Handbuch (englisch)
- Kurzbeschreibung zur EGA-Karte
- diverse Utilities, z. B. zum Übertragen von Dateien über die serielle Schnittstelle.



Bild 1. Das Gehäuse des Kaypro K286 weicht vom Standard-AT-Gehäuse ab

Insgesamt lagen dem Gerät elf Disketten (360 KByte Kapazität) – die Begleitdisketten zu den Büchern noch gar nicht gerechnet – voller Software bei. Wir haben uns bei Kaypro telefonisch vergewissert, daß alle aufgezählten Dinge wirklich zum Standard-Lieferumfang gehören und nicht nur eine zufällige Beigabe des Testgerätes waren. Man erhält mit dem Kaypro K286 wesentlich mehr als nur einen „nackten“ Computer.

Mancher Leser wird jetzt einwenden, daß die aufgezählten Zugaben keineswegs wirklich kostenlos sind, sondern der Verkaufspreis des Computers eben entsprechend höher angesetzt wurde. Dies ist nur bis zu einem gewissen Punkt richtig. Bei dem sogenannten „Bundling“, dem Beipacken von Dingen, verpflichtet sich der Hersteller, jeden Computer mit dem entsprechenden Extra auszuliefern und erhält dafür die Software oder das Buch zu einem extrem günstigen Preis aufgrund der hohen Abnahmemenge. Selbst Marken-Software kostet so oft zwischen 10 und 30 Prozent des eigentlichen Verkaufspreises.

Da der Hersteller mit seiner Hardware und nicht mit den Extras sein Geld verdient, erhält der Kunde die Zugabe meist zum Selbstkostenpreis. Auf diese Weise bekommt man für einen geringen Aufpreis das ohnehin benötigte Textverarbeitungsprogramm gleich mitgeliefert. Manche der Zugaben sind allerdings oft wertlos. So wird der deutschsprachige Käufer mit dem Word Finder, einem elektronischen Wörterbuch für englische Synonyme, nur wenig anzufangen wissen. Alle weitere Extras sind gut.

Platz für vier Laufwerke

Im Gehäuse des Kaypro K286 (*Bild 1*) kann man vier von außen zugängliche Slim-Line-Laufwerke installieren. Im Inneren des Gehäuses ist Platz für drei kurze und sechs lange Steckkarten, wobei die vier 16-Bit-Steckplätze auf lange Karten entfallen. Einer der kurzen Steckplätze zählt nur bedingt, da das Abschlußblech keinerlei Anschlüsse oder Schalter aufweisen darf.

In einem der 16-Bit-Steckplätze befindet sich die CPU-Karte des modular aufgebauten Computers. Trotz der Verwendung dynamischer 256-KBit-Speicher mit platzsparenden ZIP-Gehäusen (Zick-Zack-Inline) blieb wegen den ansonsten weitgehend in konventionelle Gehäuseformen montierten Bauelementen auf der CPU-Platine kein

NOCH EIN **PLUS.**

ORGATECHNIK KÖLN '88
INTERNATIONALE BÜROMESSE
Halle 01.2/Gang O 62/P 61
20. bis 25. Oktober



Ein weiteres Plus der erfolgreichen SANYO-PC's, **der MBC 18 Plus.** Er ist der Profi-PC für alle Anwendungen, bei denen es auf größte Schnelligkeit und Flexibilität ankommt. Die konsequente 32-bit Technologie macht ihn zum Renner unter den Arbeitsplatz-Computern. Seine Kapazität und die Schnelligkeit machen ihn zum perfekten Computer für Bereiche, in denen Perfektion und Schnelligkeit eine wichtige Rolle spielen. Desktop-Publishing, Wissenschaft und Forschung, werden mit diesem PC an Effizienz gewinnen. **SANYO MBC 18 Plus,** der kompakte PC für Anwender, die höchste Leistung und Geschwindigkeit fordern.

PERSONAL-COMPUTER-COUPON
Bitte senden Sie mir schnellstens mehr Informationen über den MBC 18 Plus.

Name, Firma

Straße

PLZ/Ort

Telefon

SANYO Büro-Electronic Europa Vertrieb GmbH
Postfach 801740 8000 München 80 Tel. 089/41604-0

MC

SANYO

...denn Qualität ist kein Luxus

Platz für die serielle und parallele Schnittstelle. Die beiden Schnittstellen befinden sich, ebenso wie die EGA-Karte und der kombinierte Floppy- und Festplatten-Controller, auf einer eigenen kurzen Steckkarte. Die CPU-Karte ist mit hochintegrierten Bausteinen von Faraday, einer Tochterfirma von Western Digital, bestückt.
Für den Anwender bleiben noch vier Steckplätze zur freien Verfügung übrig. Bei Ausnutzung aller Steckplätze und der Verwendung von vier Laufwerken könnte man mit dem 160-Watt-Netzteil unter Umständen in Schwierigkeiten kommen. Positiv an dem Netzteil ist der leise Lüftermotor, der sogar von der Festplatte und vor allem vom Diskettenlaufwerk mit seinem „sägenden" Stepper-Motor übertönt wird.

Umschaltbare Taktfrequenz

Die gemessenen Zeiten für die Benchmark-Programme PRIM1, PRIM2 [1], Sieb des Eratosthenes und Fibonacci [2] sind in *Bild 2* grafisch dargestellt. Die Werte des Kaypro K286 sind nahezu identisch mit den Ergebnissen des Schneider PC2640.
Die Umschaltung der Taktfrequenz (6/12 MHz) erfolgt über einen am Abschlußblech der CPU-Karte angebrachten Kippschalter. Einen Wechsel der Taktfrequenz bei eingeschaltetem Rechner sollte man unbedingt unterlassen, da dies in den meisten Fällen zum Systemabsturz führt. So geschah es beim Wechseln der Taktfrequenz mehrfach, daß die Tastatur nach dem Umschalten tot war. Weder ein Warmstart (Ctrl-Alt-Del) noch ein Kaltstart, der Reset-Knopf befindet sich übrigens oberhalb des Schalters für die Taktfrequenz, brachte hier Abhilfe. Als letzte Rettung blieb nur das Aus- und Einschalten des Computers.
Ansonsten traten während der dreiwöchigen Erprobungsphase keine Probleme auf. Eine Vielzahl an Programmen liefen auf Anhieb. Inkompatibilitäten sind angesichts des im Kaypro K286 verwendeten und sehr verbreiteten Phoenix-Bios nicht zu erwarten. Beim MS-DOS-Betriebssystem ist Kaypro mit der Version 3.2 zwar nicht auf dem neuesten Stand, doch sind die Unterschiede zwischen Version 3.2 und 3.3 nicht besonders aufregend. Eine der wichtigsten Neuerungen von DOS 3.3, die Aufteilung einer Festplatte mit mehr als 33 MByte in mehrere logische Laufwerke, wird ohnehin mit der Utility SpeedStor bestens ausgeglichen.
Die MF-II-kompatible Tastatur des Kaypro K286 besitzt einen mittelharten Anschlag mit einem gut fühlbaren Druckpunkt. MF-II-kompatible Tastaturen haben die Caps-Lock-Taste an der Stelle, an der sich bei den alten AT-Tastaturen die Ctrl-Taste befindet.

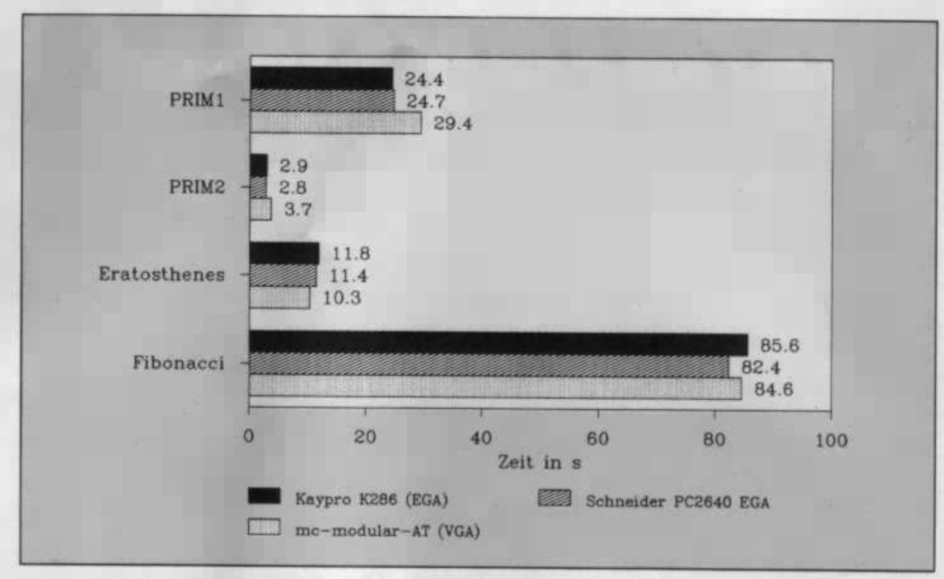


Bild 2. Mit seiner Taktfrequenz von 12 MHz liegt der Kaypro K286 mit an der Spitze

Die technischen Daten des Testgeräts

Typ	Kaypro K286
Hersteller	Kaypro Corporation
CPU	80286
Taktfrequenz	6/12 MHz (per Schalter wählbar)
Coprozessor	Fassung für 80287 vorhanden
Speicher	1 MByte auf der CPU-Platine
Diskettenlaufwerk	5¼-Zoll mit 1,2 MByte Kapazität
Festplatte	5¼-Zoll mit 20 oder 40 MByte Kapazität
Steckplätze	6 lange Steckplätze (2 × 8 Bit und 4 × 16 Bit) 2 kurze Steckplätze (8 Bit) 1 kurzer Steckplatz für Karten ohne Anschlüsse
Schnittstellen	eine parallele Schnittstelle (25pol. Buchse) und eine serielle Schnittstelle (9pol. Stecker) auf einer Zusatzkarte
Tastatur	MF-II-kompatibel
Bildschirmkarte	EGA-Karte (Half-Card)
Monitor	14-Zoll-EGA-Monitor
Betriebssystem	MS-DOS 3.21
Reset-Schalter	ja
Netzteil	160 W Leistung
Gehäuseabmessungen	490 × 420 × 130 mm (B × T × H)
Sonstiges	Festplatten-Utility SpeedStor Wordstar Version 4.0 (englisch) Word Finder (englisch) diverse Utilities

Umsteiger vertippen sich daher häufig. Bei der Tastatur des Kaypro K286 kann man mit einem DIP-Schalter auf der Unterseite der Tastatur die Funktion der Caps-Lock und der Control-Taste vertauschen. Zwei Tastkappen mit entsprechender Größe und Aufschrift werden mitgeliefert und runden die Anpassung ab.
Der zusammen mit dem Testgerät gelieferte 14-Zoll-EGA-Monitor von Kaypro mit der Bezeichnung KP-1453M1 zeichnet sich durch klare Farben und ein stabiles Bild aus. Der Monitor besitzt einen abnehmbaren Schwenkfuß. Helligkeits- und Kontrastregler sind verdeckt auf der Vorderseite angebracht.

Der Kaypro K286 gehört mit seinen 12 MHz zu den schnellsten seiner Klasse. Interessant wird das Gerät durch die Beigabe reichhaltiger und nützlicher Software und Bücher. Im Preis von 5871 DM ist die 20-MByte-Festplatte bereits inbegriffen. Vor allem wer den mitgelieferten Wordstar in der aktuellen Version 4.0, der allein schon 1470 DM kostet, gebrauchen kann, ist mit dem Kaypro K286 gut bedient.

Literatur

[1] *Dieter Strauß*: Primzahlen-Sucher. mc 1987, Ausgabe 10, Seite 74.
[2] High-Tech Horsepower. Byte Juli 1987, Seite 101.

Günther Sternberg

Ein Laptop aus zwei Teilen

Snap 1+1 von Interquadram auf dem Prüfstand

Tragbare Computer gleichen wie ein Ei dem anderen, heißt es. Daß das nicht so sein muß, beweist Interquadram mit dem Snap 1+1, der sich von der Masse der Laptops deutlich unterscheidet. mc hat den Snap 1+1 kritisch geprüft und legt seine Plus- und Minuspunkte offen.

Interquadram in Neu-Isenburg bietet mit dem Snap 1+1 (*Bild 1*) einen Laptop an, der in vielen Details von anderen Systemen abweicht. Seine Bezeichnung deutet bereits auf die Zweiteilung in Lapheld- und Erweiterungs-Modul hin. Der Anwender kann den schweren Laufwerkteil abtrennen und dennoch mit dem System weiterarbeiten, weil eine batteriegepufferte RAM-Floppy die Daten zuverlässig speichert. Genaugenommen besteht der Laptop aus drei Teilen, da man auch die LC-Anzeige abnehmen kann. Das Lapheld-Modul, der vordere Teil des Geräts mit Systemplatine, Tastatur, Anzeige und RAM-Floppy, wiegt nur vier Kilogramm. Der Snap 1+1 ist etwas ungewöhnlich, aber praktisch konzipiert.

Bei dem Laptop handelt es sich um einen getunten IBM-kompatiblen PC im Kleinformat. Er ist mit einer NEC V20-CPU bestückt. Die Taktfrequenz kann per Tastenkombination zwischen 4,77 MHz und 9,54 MHz umgeschaltet werden. Die Ergebnisse der mc-Benchmarks in *Bild 2* zeigen, daß der Snap 1+1 den Vergleich mit anderen Geräten nicht scheuen muß.

Das Testgerät enthielt 640 KByte RAM-Speicher, ein 3½-Zoll-Laufwerk mit 720 KByte Kapazität und eine 20 MByte Festplatte. Die Hintergrundbeleuchtung der Super-Twist-LCD-Anzeige erhöht die Lesbarkeit ganz erheblich. Die blaue Schrift auf hellem Untergrund ist selbst unter einem ungünstigeren Blickwinkel und bei schlechten Lichtverhältnissen gut zu lesen. Die Frage, wie sich die stromfressende Hintergrundbeleuchtung auf die Arbeitsdauer im Batteriebetrieb auswirkt, ließ sich leider nicht klären, da zum Zeitpunkt des Tests der Akkusatz noch nicht zur Verfügung stand.



Bild 1. Der Snap 1+1 ist ein etwas ungewöhnlicher Laptop

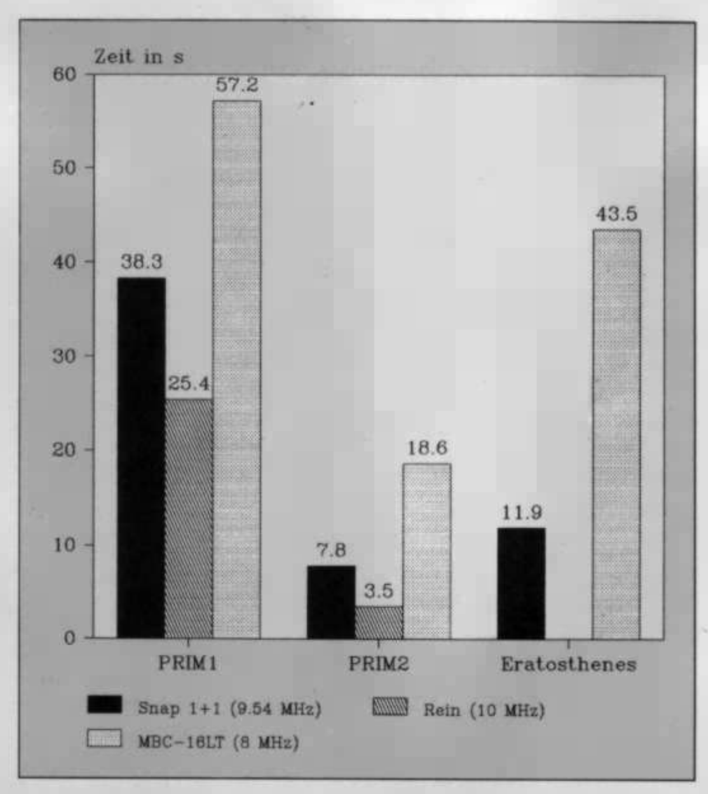


Bild 2. Die CPU erreicht beinahe die Leistungsfähigkeit von Computern der AT-Klasse

Der für die Akkus vorgesehene Raum im Inneren des Erweiterungs-Moduls fiel dem größeren Platzbedarf der Festplatte zum Opfer. Die Akkus müssen daher in einem externen Gehäuse untergebracht werden. Da der Strom der Akkus nicht ausreicht, um die Festplatte zu betreiben, ist man sowieso auf das Netzgerät angewiesen.

Platz für zahlreiche Einschübe

Die Liste der externen Anschlüsse und Einschübe für Zusatzkarten ist beeindruckend. Das Lapheld-Modul kann eine Speichererweiterungskarte mit wahlweise 500 KByte oder 1 MByte statischen Speichers mit Batteriepufferung und eine Erweiterungskarte für ein Hayes-kompatibles Modem aufnehmen. Zur Installation der Karten muß man die beiden Teile des Laptops durch Öffnen der vier Schrauben (als Schraubendreher genügt eine Münze) trennen. Die Karten schiebt man darauf in die entsprechenden Steckplätze. Auf der linken Seite des Lapheld-Moduls sind die Anschlüsse für das Modem und die serielle Schnittstelle (25poliger D-Stecker) sowie der Reset-Knopf und der Kontrastregler der LCD-Anzeige untergebracht. Auf der rechten Seite befinden sich die Anschlüsse für die parallele Schnittstelle und den externen Monitor. Der Monitor läßt sich entweder über eine neunpolige RGB-Buchse oder eine Composite-Video-Buchse anschließen. Ebenfalls an der rechten Seite befindet sich der Einschalter des Geräts.

Auf der Rückseite des Erweiterungsmoduls findet man den

Die technischen Daten der Module auf einen Blick

Das Lapheld-Modul
- NEC V20-CPU mit 4,77-MHz- oder 9,54-MHz-Takt
- 640 KByte Hauptspeicher
- optionelle Speichererweiterung mit 512 KByte oder 1 MByte statischen RAM mit Batteriepufferung
- Super-Twist-LCD mit 640 × 200 Pixel und Hintergrundbeleuchtung
- RAM-Floppy mit maximal 720 KByte Kapazität
- Tastatur mit 84 Tasten
- serielle Schnittstelle (RS232)
- parallele Schnittstelle (Centronics)
- RGB-Anschluß für externen Monitor
- Composite-Video-Anschluß für externen Monitor
- Steckplatz für Modem-Karte (Hayes-kompatibel)
- Kontrastregler für LCD-Anzeige
- Reset-Knopf
- Echtzeituhr
- Gewicht ca. 4,2 kg
- Maße 19 × 33 × 7 cm^3

Das Erweiterungs-Modul
- zwei 3½-Zoll-Laufwerke (720 KByte) oder ein Laufwerk und eine 3½-Zoll-Festplatte (20 MByte)
- Anschluß für externes 5¼-Zoll-Laufwerk (360 KByte)
- ladbare NiCd-Akkus für ungefähr sechs Stunden Betriebsdauer (bei Modell mit Festplatte extern)
- Slot für kurze Steckkarte
- Tragegriff, der zugleich als Aufstellbügel dient
- Gewicht ca. 4,8 kg
- Maße 14 × 33 × 7 cm^3

Anschluß für das externe 5¼-Zoll-Laufwerk und die Buchse für das Netzgerät. Damit nicht genug, ein Steckplatz für eine kurze Erweiterungskarte ist auch noch vorhanden. Auf der linken Seite des Moduls ist das erste Laufwerk angebracht, auf der rechten Seite die Festplatte oder das zweite Laufwerk. Den Abschluß bildet der bei dieser Packungsdichte wohl unvermeidliche Lüfter auf der Rückseite.
Alle Anschlüße und Steckplätze sind vorbildlich mit Plastikklappen abgedeckt, die aber allesamt ein wenig klemmen. Ansonsten macht das dunkelgraue Spritzgußgehäuse einen recht stabilen Eindruck. Übrigens, der Tragegriff ist an der Rückseite des Geräts angebracht und dient gleichzeitig als Aufstellbügel. Ein Nachteil durch die seitliche Anordnung der Anschlüße sei nicht verschwiegen. Benutzt man zum Beispiel die serielle und parallele Schnittstelle, stehen die Kabel links und rechts vom Gerät ab und sind ständig im Weg.

Das Gerät im praktischen Betrieb

Bei dem Snap 1+1 scheint man besonders viel Mühe für die Entwicklung eines komfortablen BIOS aufgewendet zu haben. So erscheint nach jedem Kaltstart zunächst eine Seite mit den aktuellen Systemeinstellungen (Setup). Durch Betätigung einer beliebigen Taste innerhalb von ca. 2 Sekunden lassen sich die verschiedenen Werte ändern. Einige der per Menü einstellbaren Parameter sind

- die Größe der RAM-Floppy
- Booten von der RAM-Floppy
- die Taktfrequenz (4,77 oder 9,54 MHz)
- die Anzahl der Laufwerke

und vieles mehr.

Nach dem Setup werden die Größe des Hauptspeichers, des Erweiterungsspeichers und der RAM-Floppy sowie Datum und Uhrzeit der batteriegepufferten Echtzeituhr angezeigt.
Auch während des Betriebs stehen gut ein Dutzend, über bestimmte Tastenkombinationen aufrufbare, Zusatzfunktionen zur Verfügung. So kann man beispielsweise zur Schonung der Akkus festlegen, daß bei einer Eingabepause (Zeitspanne 30 Sekunden bis 60 Minuten) die Anzeige automatisch abgeschaltet wird.
Im Laufe der vierwöchigen Testphase wurde das Gerät intensiv genutzt. So wurde unter anderem ein vierzigseitiger Bericht mit Euroscript eingegeben und bis zur Druckreife bearbeitet. Dabei fiel vor allem die Tastatur angenehm auf. Die Tasten besitzen einen angenehmen Druckpunkt und quittieren jede Betätigung mit einem mechanischen Klick.
Die RAM-Floppy läßt sich wahlweise als 360 KByte oder 720 KByte Laufwerk einrichten, so daß auch das Booten von RAM-Floppy kein Problem darstellt. Durch eigene Pufferbatterien bleiben die gespeicherten Daten ungefähr einen Monat lang erhalten. Zusätzliche Akkus auf der Speicher-Erweiterungskarte liefern den Betriebsstrom für das Lapheld-Modul. Ohne Erweiterungskarte oder für längere Arbeiten muß man das externe Netzgerät als Stromspender anschließen.
Die durch die Zweiteilung mögliche Gewichtsersparnis von rund 5 kg macht den Snap 1+1 zum idealen Partner für unterwegs – vorausgesetzt, man kommt ohne Laufwerk und Festplatte aus.

Verbesserungen erwünscht

Wie fast alle Computer, läßt sich auch der Snap 1+1 noch verbessern. Dem Testgerät waren die deutschen Tastaturkappen in einer kleinen Tüte beigefügt. Ein Beiblatt, das die Montage der Tastaturkappen erklärt, war dem Gerät nicht beigefügt. Im deutschen Tastaturtreiber sind die Codes für Control-Y und Control-Z vertauscht. Außerdem kann man keinen einzelnen Apostrophen, sondern nur Akzentbuchstaben eingeben. Auch die MS-DOS-Version ist altertümlich. Es handelt sich um MS-DOS Version 2.11 mit einer Anpassung auf 3½-Zoll-Laufwerke. Da es sich hier um leicht zu behebende Fehler in der System-Software handelt, dürften sie bis zum Erscheinen des Testberichts behoben worden sein.

Positiver Gesamteindruck

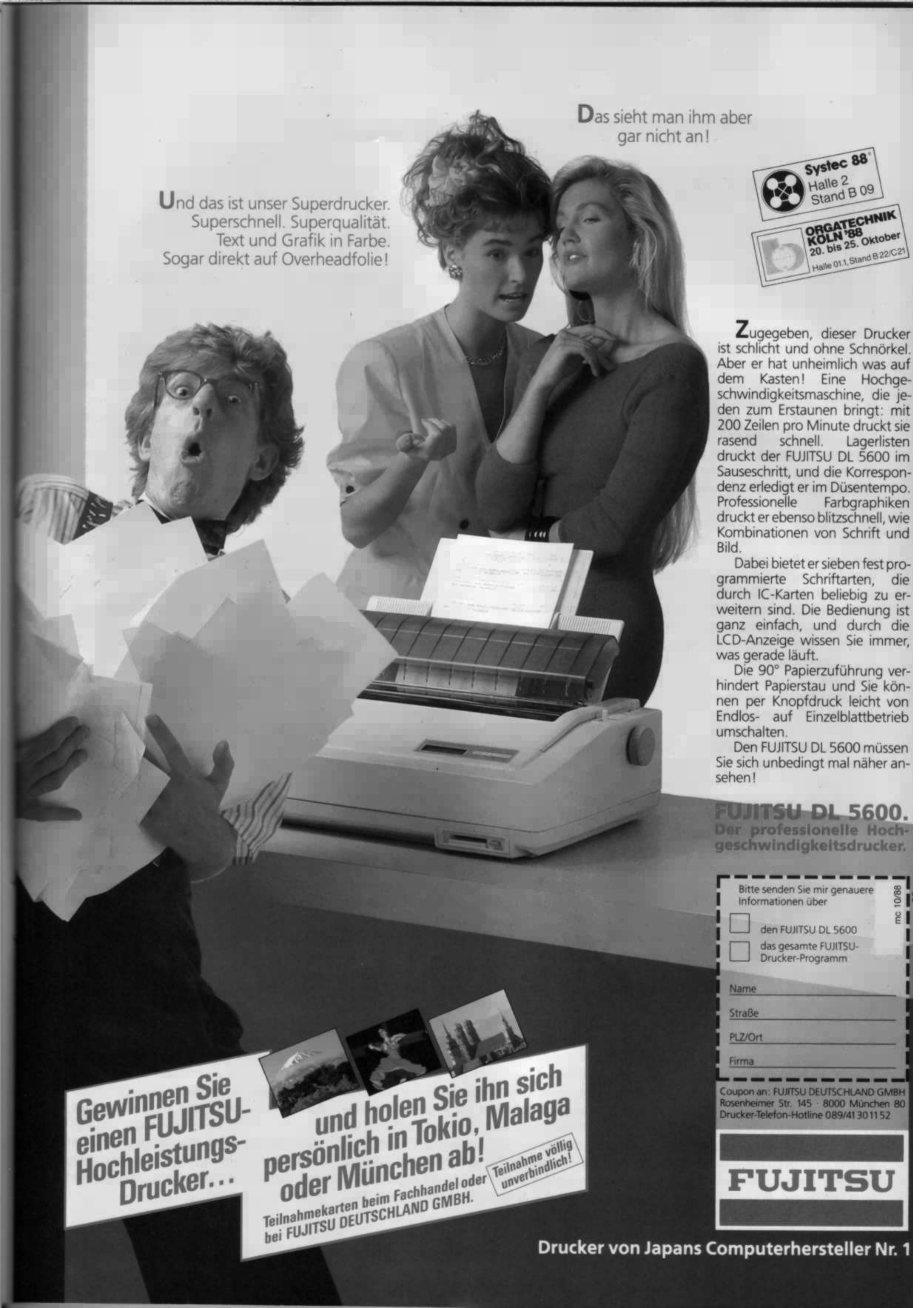
Die vielen nützlichen Details und der Probebetrieb hinterlassen einen guten Gesamteindruck. Wegen dem externen Akkusatz und dem externen Netzteil ist das System etwas unhandlich. Die beiden Handbücher sind zwar sehr ausführlich und beschreiben den Umgang mit Computer und Betriebssystem, doch leider in englischer Sprache. Im Preis von etwas unter 8000 DM ist die 20-MByte-Festplatte bereits inbegriffen. □

Spruch des Monats

In Zukunft besteht die Arbeit nicht mehr darin, seinen Lebensunterhalt zu verdienen, sondern darin, im Zeitalter der Automation leben zu lernen.

Marshal McLuhan

gefunden in
„Das Chaos Computer Buch", Rowohlt Verlag.


Das sieht man ihm aber gar nicht an!
Und das ist unser Superdrucker. Superschnell. Superqualität. Text und Grafik in Farbe. Sogar direkt auf Overheadfolie!
Systec 88
Halle 2
Stand B 09
ORGATECHNIK KÖLN '88
20. bis 25. Oktober
Halle 01.1, Stand B 22/C21
Zugegeben, dieser Drucker ist schlicht und ohne Schnörkel. Aber er hat unheimlich was auf dem Kasten! Eine Hochgeschwindigkeitsmaschine, die jeden zum Erstaunen bringt: mit 200 Zeilen pro Minute druckt sie rasend schnell. Lagerlisten druckt der FUJITSU DL 5600 im Sauseschritt, und die Korrespondenz erledigt er im Düsentempo. Professionelle Farbgraphiken druckt er ebenso blitzschnell, wie Kombinationen von Schrift und Bild.
Dabei bietet er sieben fest programmierte Schriftarten, die durch IC-Karten beliebig zu erweitern sind. Die Bedienung ist ganz einfach, und durch die LCD-Anzeige wissen Sie immer, was gerade läuft.
Die 90° Papierzuführung verhindert Papierstau und Sie können per Knopfdruck leicht von Endlos- auf Einzelblattbetrieb umschalten.
Den FUJITSU DL 5600 müssen Sie sich unbedingt mal näher ansehen!
FUJITSU DL 5600.
Der professionelle Hochgeschwindigkeitsdrucker.
Bitte senden Sie mir genauere Informationen über
mc 10/88
den FUJITSU DL 5600
das gesamte FUJITSU-Drucker-Programm
Name
Straße
PLZ/Ort
Firma
Coupon an: FUJITSU DEUTSCHLAND GMBH
Rosenheimer Str. 145 · 8000 München 80
Drucker-Telefon-Hotline 089/41 30 11 52
FUJITSU
Gewinnen Sie einen FUJITSU-Hochleistungs-Drucker...
und holen Sie ihn sich persönlich in Tokio, Malaga oder München ab!
Teilnahme völlig unverbindlich!
Teilnahmekarten beim Fachhandel oder bei FUJITSU DEUTSCHLAND GMBH.
Drucker von Japans Computerhersteller Nr. 1

Dipl.phil. Lore Schönrock

Die Star-Parade

Star-Writer PC 3.0, Star-Planer PC, Star-Manager PC

Die Star-Division entwickelt ihre Software in Deutschland, um den hohen Ansprüchen der deutschen Anwender gerecht zu werden. Eine Palette preiswerter und hochwertiger PC-Software wird dem privaten Anwender zugänglich gemacht und auch einem beruflichen Einsatz gerecht.

Der Personal Computer ist im Laufe der letzten Jahre zum alltäglichen Werkzeug zur Erledigung beruflicher und privater Aufgaben geworden. Mit verschiedener Software werden ebenso verschiedene Aufgaben bewältigt. Der Computer wird beispielsweise für die Buchhaltung, Lagerverwaltung und Korrespondenz benutzt. Meist wird sogar an einem PC mit Festplatte mit mehreren unterschiedlich zu bedienenden Softwarepaketen gearbeitet.

Als vielseitiges und zuverlässiges Arbeitsmittel muß der PC entsprechend gepflegt und die Arbeit organisiert werden. Beispielsweise sollten die verschiedenen Programme in unterschiedlichen Verzeichnissen auf der Festplatte geführt werden, um eine gewisse Ordnung und Übersicht zu wahren. In bestimmten zeitlichen Abständen muß die Festplatte aufgeräumt werden, um Platz zu schaffen. Dazu werden die Dateien gesichtet, einige gelöscht und andere auf Disketten ausgelagert.

Für diese Arbeiten kann man das Betriebssystem MS-DOS mit seinen dafür gedachten Funktionen einsetzen. Doch stammt das MS-DOS noch aus einer Zeit der Computerentwicklung, in der dem Anwender einiges an Wissen über das Betriebssystem, über die Vielzahl seiner Funktionen und Parameter abverlangt wurde. Der Anwender muß, um seine Programme und Daten unter MS-DOS verwalten und organisieren zu können, die englischen Befehlskürzel und die Programm-Syntax des Betriebssystems erlernen. Beim Einsatz dieser Befehle kommt meist eine nicht unerhebliche und fehlerträchtige Tipparbeit hinzu.

Heutzutage ist die Bedienerfreundlichkeit der Computer in aller Munde. Die Steuerung der Anwenderprogramme wird bereits durch Menüs vereinfacht oder erfolgt grafisch über Symbole. Neben der Cursor-Tastatur kann häufig auch die Maus zur Auslösung der Funktionen verwendet werden.

Viele Hersteller denken nun auch daran, eine einheitliche Benutzeroberfläche für die Verwaltungs- und Organisationsarbeiten der aufgenommenen Daten anzubieten. Für diese Aufgaben ist zwar das Betriebssystem zuständig, doch hat sich an dessen Benutzeroberfläche, an der Art und Weise wie die Informationen am Bildschirm dargestellt werden und auszulösen sind, nichts geändert. Erst die neueste Version MS-DOS 4.0 wartet mit einer grafischen Benutzeroberfläche auf.

Der Star-Manager PC ist eines von drei Programmen der Star-Division, die wir getestet haben. Er bietet eine anwenderfreundliche, einfach zu bedienende Benutzeroberfläche zur Handhabung von Daten auf Festplatte und Disketten. Das Programm nutzt jedoch nicht nur alle Funktionen des Betriebssystems MS-DOS. Vielmehr wurden ein Editor, ein „Taschen-" Rechner, eine ASCII-Tabelle, ein Karteikasten und eine Terminverwaltung im gleichen Programm integriert.

Vom Star-Manager aus kann jedes auf der Festplatte installierte Programm gestartet werden. Die Voraussetzung für seinen Einsatz ist ein zum IBM-PC kompatibler Rechner mit mindestens 384 KByte RAM und MS-DOS 3.x. Obgleich der Star-Manager PC für Festplattengeräte konzipiert ist, läßt er sich mit etwas Mehraufwand auch auf Rechnern mit nur einem Diskettenlaufwerk nutzen.

Den Star-Manager PC kann man sowohl mit der Tastatur als auch mit der Maus steuern. Das Programm erkennt automatisch, daß eine Maus an den Computer angeschlossen ist und stellt einen Pfeil als Mauszeiger am Bildschirm dar. Unter dem Menü „Optionen" wird sowohl die Farbe wie auch ein anderes Zeichen für den Mauszeiger ausgewählt.

Am oberen Rand des Bildschirmes steht eine Menüzeile, aus der weitere Untermenüs erreichbar sind. Darunter sieht man die Bezeichnung der Laufwerke, die zur Auswahl stehen, wobei der Name des aktuellen Laufwerkes invers dargestellt wird. Rechts stehen der aktuelle Pfad und die Uhrzeit. Unter den Laufwerksbezeichnungen wird eine Liste mit den Dateien und Unterverzeichnissen im aktuellen Pfad dargestellt. Über der Menüleiste und dem beschriebenen Grundbild steht der Name und die Version des Programmes.

In der Menüzeile sind folgende Auswahlpunkte aufgeführt: Datei, Platte, Funktionen, Optionen, Anwendungen. Jeder dieser Punkte vereint in seinem Untermenü die entsprechenden Funktionen. Die einzelnen Untermenüs werden in Fenstern dargestellt.

Attribute ändern – Festplatten optimieren

Auf die englischen Kürzel, die die Befehle in MS-DOS ausmachen, ist zugunsten deutscher Benennungen verzichtet worden. So sind unter der Überschrift „Datei" die zur Handhabung von Dateien notwendigen Funktionen zusammengefaßt. Hierüber lassen sich die Verzeichnisse aufrufen, eine Datensicherung vornehmen, Dateien löschen, umbenennen oder kopieren.

Über die Standard-Funktionen von MS-DOS hinaus, stellt das Programm weitere nützliche Funktionen zur Verfügung. Sei es das Umbenennen von Verzeichnissen oder das alphabetische Sortieren eines Verzeichnisses nach Namen, Erweiterung oder Dateigröße – der Star-Manager PC bietet die richtige Funktion.

Wenn nach einer bestimmten Datei gesucht wird, können Wildcards (Ersetzungszeichen) verwendet werden. Das Programm sucht im ganzen Verzeichnis des aktuellen Laufwerkes und zeigt jede gefundene Datei, deren Bezeichnung mit dem Suchbegriff übereinstimmt, mit dem dazugehörenden Pfadnamen an.

Ist eine Datei gelöscht und danach noch kein neuer Text gespeichert worden, kann der Manager diese Datei wiederherstellen. Sind bereits neue Daten gesichert worden, kann die Datei nur teilweise wiederhergestellt werden. Der Star-Manager PC gibt dann eine Meldung über die Dateigröße und darüber, wieviele Byte noch gerettet werden können.

Im Unterpunkt „Information" kann gezielt Auskunft über die Dateigröße, dem auf dem Datenträger belgten Speicherplatz sowie über die Nummer des ersten Daten-Clusters der Datei, Datum und Uhrzeit der jüngsten Änderung und Attribute der Datei angefordert werden.
Die Attribute weisen auf die Art der Datei hin (x.hid steht beispielsweise für eine versteckte Datei). Die Attribute kann man ändern, bei Bedarf auch das Datum und die Uhrzeit. Für diese Arbeit lassen sich auch die Dateien, die die gleichen Änderungen erhalten sollen, zusammenfassen. Vier Attribute werden dazu angeboten:
ARC – Seit dem letzten Backup-Lauf ist diese Datei geändert worden und wird beim nächsten Backup-Lauf erneut archiviert.
HID und auch SYS – Versteckte Dateien werden gewöhnlich nicht angezeigt. Sie erscheinen nur in der gewählten Verzeichnisanzeige.
RO – Schreibgeschützte Dateien dürfen nur gelesen, nicht bearbeitet werden.
Aus dem Menü „Datei" lassen sich eine oder mehrere Dateien für den Druck bestimmen. In einem Fenster wird die im Druck befindliche Datei mit Pfad und Laufwerksangabe angezeigt. Der Ausdruck kann abgebrochen werden.
Zum Unterpunkt „Verzeichnisse" gehört die grafische, baumförmige Übersicht über die Verzeichnisse mit Unterverzeichnissen. Daraus lassen sich Unterverzeichnisse von einem Ast des Baumes entfernen und an einen anderen anhängen. Somit läßt sich die Struktur einer Festplatte oder Diskette jederzeit den aktuellen Erfordernissen anpassen.
Arbeitsgänge, die für die Verwaltung kompletter Disketten oder Festplatten gebraucht werden, sind unter der Bezeichnung „Platte" zu finden. Die Datenträger lassen sich unter anderem prüfen, formatieren und optimieren. Störend an diesem Programmteil ist, daß auch Disketten als „Platten" bezeichnet werden.
Die Funktion zur Optimierung der „Platten" verlegt alle Verzeichnisse in die ersten Sektoren des Datenträgers. Dabei werden alle Dateien sequentiell abgelegt. Ihre Sektoren sind damit hintereinander angeordnet, wodurch sich die Zugriffszeiten verkürzen. Versteckte Dateien und Systemdateien sind davon nicht betroffen – sie bleiben unverändert.

Editor für Programmierer

Im Menü „Funktionen" befindet sich der Editor. Damit lassen sich vier unterschiedliche Dateien ansehen und auch ändern. Der Texteditor arbeitet auf Wunsch mit einem WordStar-kompatiblen Befehlssatz. Damit lassen sich Dateien von der Platte ansehen und ändern, auch Programme schreiben oder Notizen anfertigen.

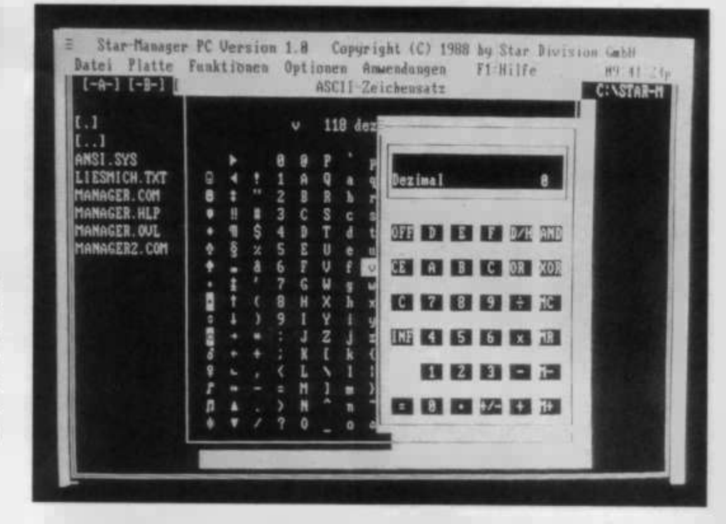


Bild 1. Für kleine Rechnungen ist der integrierte Taschenrechner vorgesehen – sicherlich eine hilfreiche Funktion

Darüber hinaus verfügt das Programm über einen Cluster-, Sektoren- und einen RAM-Editor. Der Editor stellt den Speicherinhalt als ASCII-Text oder als hexadezimal Zahlen dar. Beim Aufruf des Cluster-, Sektor- und RAM-Editors arbeitet der Editor gleich im Hex-Modus. Desgleichen auch beim Aufruf der Dateien mit den Endungen COM, EXE, SYS. Mit dieser Editierfunktion können sowohl Konfigurationsdateien von Programmen eingesehen als auch bestimmte Speicherbereiche auf der Festplatte oder im PC.
Der Taschenrechner des Star-Manager PC kennt die vier Grundrechenarten. Steht der Cursor beim Aufruf dieser Funktion in einem Editierfeld, so wird nach Abschluß der Rechnung das Ergebnis an diese Cursorposition übernommen (*Bild 1*).
Aus der ASCII-Tabelle lassen sich z. B. Angaben für den Editor kopieren.
Unter der Funktion „Karteikasten" werden Notizen erfaßt, bearbeitet und abgelegt. Sie müssen die Erweiterung .CRD tragen, um über die Funktion eingelesen werden zu können. Es kann nach Karteikarten gesucht werden. Karteikarten lassen sich einzeln oder auch alle auf einmal ausdrucken.
Der Terminkalender ist mit einfachen Funktionen für die Planung und Einhaltung von Terminen ausgestattet. Sie sind über die Funktionstasten auszulösen. Der Computer erinnert durch ein akustisches Signal an Termine.
Wenn man im Menü „Optionen" den WordStar-Modus eingeschaltet hat, läßt sich eine Eintragung mit den WordStar-Tastenfunktionen markieren, löschen, oder an eine andere Stelle kopieren oder verschieben. Zwei Darstellungen werden angeboten: Die Tages- und die Monatsansicht. In der Monatsübersicht lassen sich wichtige Termine mit Ausrufezeichen versehen, um sofort gesehen zu werden. Für die eingebaute Weckfunktion ist ein eigenes Fenster zur Eingabe der Weckzeit vorgegeben. Wenn die interne Uhr die Weckzeit erreicht hat, wird alle 10 Sekunden ein Piepston ausgelöst und eine Meldung in einem Textfenster eingeblendet.
Unter dem Menüpunkt „Anwendungen" lassen sich andere Anwendungsprogramme in das Menü-System von Star-Manager PC integrieren, so daß sie ein Bestandteil vom Star-Manager PC werden. Neue Anwendungen können eingefügt, geändert und auch wieder gelöscht werden. Die integrierten Programme werden im Pull-Down-Menü ausgewiesen und lassen sich von hier aus starten.

Star-Writer PC 3.0

Das Programm Star-Writer PC bietet Textverarbeitung, Adressverwaltung, ein Grafik- und ein Datenfernübertragungsprogramm. Um diese Vielfalt nutzen zu können, sollte ein IBM-kompatibler Rechner mit 640 KByte RAM und 20 MByte-Festplatte eingesetzt werden. Darüber hinaus bedarf es einer Grafikkarte wie MDA, CGA, Hercules oder EGA, einer Maus, einem Modem zur Datenfernübertragung und einem grafikfähigen Drucker. Das Programm bietet mehr als 100 Druckertreiber zur Wahl an. So ausgestattet ist Star-Writer PC 3.0 in der Lage, die gängigen Funktionen der Textverarbeitung zu leisten und Funktionen wie Fuß- und Endnotenverwaltung, deutsche Rechtschreibkorrektur, Silbentrennung, Grafikeinbindung, Seitenlayout, Serienbrief, integrierte Datenbank, DFÜ.
Der Einsatz auf einem PC mit nur einem Diskettenlaufwerk und 256 KByte RAM ist wenig sinnvoll, da hier mit fünf ständig zu wechselnden Disketten gearbeitet werden müßte.



Bild 2. Zur Wahl stehen im Star-Writer PC neben der Textverarbeitung auch weitere Programmteile

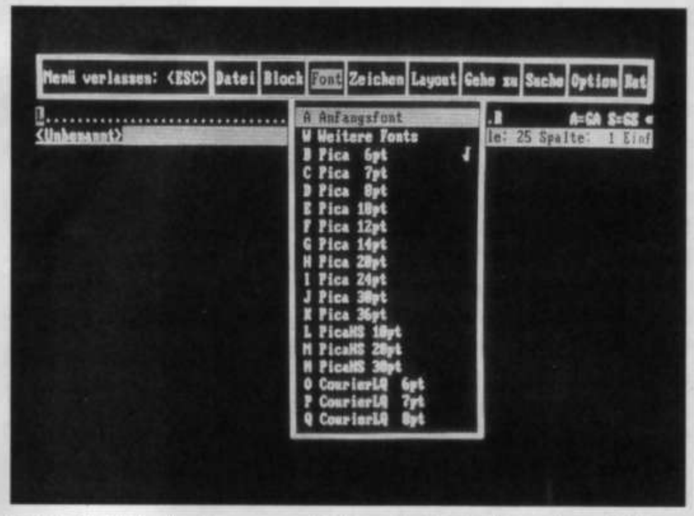


Bild 3. Die Schriftart kann sowohl über die Menüleiste wie auch über eine Tastenkombination bestimmt werden

Nach dem Programmstart muß man die gewünschte Anwendung wählen. Zur Wahl stehen Adressverwaltung, Textverarbeitung, Grafikeditor, Datenfernübertragung, Installation (*Bild 2*).

In der Textverarbeitung wird der Bildschirm zur Texteingabe freigegeben. Nur die Menüleiste, das Zeilenlinieal und der Dateiname sowie die Angaben zur Seite, Zeile, Spalte und zum Eingabemodus stehen zur Orientierung am oberen Bildschirmrand bereit. Darunter blinkt der Cursor, und bei der Texteingabe rutscht die Doppellinie, die das Textende markiert, weiter zum unteren Bildschirmrand. Sie soll das Schreibfeld nach unten abgrenzen.

Formgebung mit Layout-Dateien

ine Besonderheit der Darstellung des Textes fällt bei der Eingabe sofort auf. Star-Writer PC 3.0 orientiert sich nicht am Zeilenlinieal, sondern daran, wieviele Zeichen der ausgewählten Schrifttype in eine Druckzeile passen. Bei der Grundinstallation wird die schmalste Schrift, die der Drucker produzieren kann, als Grundschrift übernommen und über den rechten Rand hinausgeschrieben. Um diese Voreinstellung zu ändern, muß man sich gleich mit einer weiteren Besonderheit des Programmes auseinandersetzen: Mit dem Konzept der Layout-Dateien. In diesen Layout-Beschreibungen werden alle Informationen über die Form des Textes wie Seitenränder, Zeilenabstand, Tabulatoren, Einzug usw. festgeschrieben. Zu jeder Textdatei muß eine Layout-Datei existieren. Eine Layout-Datei kann für verschiedene Textdateien benutzt werden. Wenn man z. B. ein Briefformat in einer Layout-Datei festgelegt hat, kann man diese Form für die gesamte Korrespondenz nutzen. In der Layout-Datei werden alle Parameter in Ebenen festgehalten. In der ersten Ebene sind dies die Papierbreite und die Papierlänge (z. B. die des Formates A4). In der zweiten Ebene – innerhalb des beschriebenen Stück Papiers – wird das Seitenlayout definiert. Das Seitenlayout bestimmt den bedruckbaren Bereich auf dem Bogen Papier. Hier werden unter anderem die Flächen für Kopf- und Fußtexte vorgegeben oder auch der Mehrspaltendruck. In der dritten Ebene sind die Absatzformate einzugeben wie Ränder, Abstände und Einzüge, Schrift, Tabulatorstopps.

Diese Angaben sind über die Menüleiste in dafür vorgesehene Masken einzuschreiben. Die Anzahl der zur Verfügung stehenden Optionen spricht für die komplexen Gestaltungsvarianten, aber nicht für die Arbeit mit Pull-Down-Menüs. Eine der Schwäche von Pull-Down-Menüs wird an dieser Stelle ganz deutlich. Derjenige Anwender, der professionell mit dem Programm arbeitet, wird es bald als lästig empfinden, sich einen Weg durch die Menü-Wälder bahnen zu müssen. Aus der dem Software-Paket beigelegten Referenzkarte kann dieser Anwender die den einzelnen Menüpunkten entsprechenden CONTROL-Tastenkombinationen entnehmen.

Will man z. B. die zweite Schriftart des Druckers wählen, braucht man nicht den Weg über die Menüleiste zu gehen, sondern kann sich mit Ctrl-PGB direkt durchtasten (*Bild 3*).

Oder soll beispielsweise ein Text fett gedruckt werden, kann man statt in der Menüleiste unter dem Menüpunkt „Zeichen Fett" auszuwählen und damit den Beginn auszulösen und über den gleichen Weg zu beenden, beide Male die CONTROL-Tastenkombination CTRL-PN wählen oder auch den Menübefehl F10-ZN.

Im Text selbst werden die Absatz- und auch Seitenlayouts für verschiedene Textseiten durch Punktbefehle aktiviert. Auf diese Art und Weise lassen sich auch gerade und ungerade Seiten unterschiedlich gestalten. Für die automatisch einsetzbaren Seiten-, Fußnotennummern oder Daten aus den Serienbriefdateien werden Platzhalter angewendet, die das Einbinden der entsprechenden Angaben beim Druck verursachen. So ist beispielsweise für die Seitennumerierung der Kreuzbefehl S zuständig.

Es gilt generell, daß vor dem Ausdruck – gleich ob im Vorder- oder Hintergrund – der Text gespeichert werden muß. Beim Aufrufen dieser Funktion wird nicht darauf hingewiesen – der Drucker tut nichts und der Text bleibt zur weiteren Bearbeitung am Bildschirm stehen.

Starke Funktionen

Neben den gängigen Funktionen der Textverarbeitung bietet das Programm nicht nur Fuß- und Kopfzeile an, sondern auch Stichwort- und Inhaltsverzeichnis. Dazu werden die Einträge für die Verzeichnisse mit Steuerzeichen markiert. Sie sind am Bildschirm sichtbar. Beim Aufruf der Verzeichnisfunktion wird der Text dann nach diesen Steuerzeichen durchsucht und mit der Seitenzahl in eine separate ASCII-Datei geschrieben. Anschließend wird diese neue Datei alphabetisch sortiert. Sie läßt sich in ein Star-Writer-Dokument einlesen und überarbeiten.

Das Programm leistet auch Hilfe bei der Rechtschreibkorrektur und Silbentrennung, wobei es doch immer dem Anwender überlassen bleibt, die Richtigkeit zu prüfen. Trennvorschläge lassen sich bereits beim Schreiben in ein Wort einfügen. Dazu wird CONTROL-Bindestrich an die gewünschte

Stelle gesetzt. Der einfache Bindestrich dient für feste Wortverbindungen.
Zum Lieferumfang gehört ein Hauptwörterbuch. Doch können darüber hinaus ein Benutzerwörterbuch und ein Textwörterbuch für spezifisches und häufig auftretendes Vokabular angelegt werden. Bei der Überprüfung des Textes werden zunächst alle drei Wörterbücher eingesehen bevor ein Wort als falschgeschrieben gemahnt wird.
Immer wiederkehrende Redewendungen lassen sich mit der Floskelfunktion in den Floskelspeicher schreiben. Solche Wendungen können sein „Sehr geehrter Herr", „mit freundlichen Grüßen" oder auch Kreuzbefehle zum Aufruf von Variablen. Ebenso lassen sich auch Befehlsfolgen, die öfter benötigt werden, hier abspeichern. Die Funktion wird über „Optionen" angewählt. Per Menü oder mit je zwei Tastendrücken lassen sich die Floskeln auch wieder in den Text lesen. In einer Floskeldatei können 26 Floskeln – eine für jeden Buchstaben unseres Alphabets – abgespeichert werden.
Die Funktion Floskel kann das Schreiben von Briefen erleichtern, ebenso wie eine integrierte Datenbank.

Datenbank integriert

In einer integrierten Datenbank können alle wichtigen Daten in einer festzulegenden Folge aufgenommen und gespeichert werden. Die Datenbank in Star-Writer gibt eine leere Bildschirmmaske vor. Von der 4. bis zur 21. Zeile dürfen mindestens drei und höchstens 50 Felder definiert werden. Da Feldbezeichnung und Feldinhalt nicht mehr als 79 Zeichen lang sein dürfen, gilt es vor der Eingabe darüber nachzudenken, welche Angaben nötig sind.
Folgendes ist beim Ablegen der Datendateien auf Disketten zu beachten: Die Länge eines Datensatzes hängt von der Anzahl und Größe der Felder ab. Die Länge eines Datensatzes ist über Datei-Info zu ermitteln. Die Größe der Indexdatei richtet sich nach der Datenmenge und den Indexwerten. Indexdateien betragen (aus computerspezifischen Gründen) ein Vielfaches von 1024 Byte. Im Überschlag faßt eine Diskette von 360 KByte etwa 800 Datensätze.
Aufgrund der flexiblen Gestaltung lassen sich mit Star-Writer PC 3.0 nicht nur Adressdateien definieren, sondern auch Dateien zur Erfassung von Buchtiteln, Videoaufzeichnungen u.ä. Für die Gestaltung von Serienbriefen sind oftmals neben der Anschrift auch andere spezifische Daten von Bedeutung. Sie können bereits adressbezogen in der Datenbank stehen. Im Brief muß an der entsprechenden Stelle die Feldbezeichnung als Platzhalter eingesetzt werden, damit der Eintrag beim Ausdruck vorgenommen werden kann. Die Geschwindigkeit und Genauigkeit der Sortierung wird durch die Vergabe eines Indexfeldes beeinflußt. Es dürfen mindestens eins und höchstens acht Indexfelder benannt sein. Zu jedem indizierten Feld wird eine Indexdatei angelegt. Je mehr Indexfelder benannt sind, desto mehr Speicherplatz wird beansprucht. Für eine Adressverwaltung beispielsweise sind drei Indexfelder durchaus ausreichend: Name, Vorname, Postleitzahl. Damit lassen sich die gewünschten Adressen im vorgegebenen Postbereich anschreiben.

Anforderungen

Um die angebotene Vielfalt aller drei Star-Programme optimal nutzen zu können, sollte ein IBM-kompatibler Rechner mit 640 KByte RAM und 20 MByte-Festplatte eingesetzt werden sowie eine Grafikkarte wie CGA oder EGA, eine Maus, ein Modem zur Datenfernübertragung und ein grafikfähiger Matrix-, Tintenstrahl- oder Laser-Drucker.

Für eine Serienbriefaktion müssen die Daten aus der Adressverwaltung jedoch aufbereitet werden, damit sie im Programmteil Textverarbeitung auch verarbeitet werden können. Im Menü „Selektieren" gibt es dafür eine Funktion Mail-Liste. Hier erscheint die Selektionsmaske, in der einzugeben ist, welche Adressen selektiert und in die Mail-Liste aufgenommen werden sollen. Es werden generell alle Felder des Datensatzes übernommen. Beim Ausdruck des Serienbriefes werden Platzhalter durch die benötigten Daten aus der Mail-Liste ersetzt. Nach diesem Grundprinzip lassen sich Listen und Etiketten oder auch Umschläge vorbereiten und als „Serienbrief" ausdrucken.
Für kleine Rechnungen ist der integrierte Taschenrechner vorgesehen und sicherlich eine hilfreiche Funktion. Er beherrscht die einfachen Rechenarten wie Addition, Subtraktion, Multiplikation und Division sowie die Hierarchie der algebraischen Operationen. Auch Prozentrechnen und Wurzelziehen sind eingebaut.
Der Grafik-Editor des Star-Writer PC arbeitet ausschließlich im Grafikmodus und ist für die Bedienung mit einer Maus gedacht. Die Mausbewegungen lassen sich über die Cursortasten im Zahlenblock simulieren. Es handelt sich um einen punkt- oder pixelorientierten Editor. CGA, EGA und die Hercules-Karte werden unterstützt. Durch das Grafikprogramm führen Bildsymbole.
Das DFÜ-Programm des Star-Writer PC 3.0 dient zum Senden und Empfangen von Dateien. Die Baudrate ist in weiten Grenzen einstellbar. Vom Minimalwert von 110 Baud aus kann man bis zu 115200 Baud gehen. Das wird wohl nicht jede PC-PC-Verbindung mitmachen. Das DFÜ-Programm bietet den Halbduplex-Modus an. Das Programm erlaubt die Einstellung der üblichen Parameter. Darüber hinaus lassen sich die Zeilenschaltungen sowie Ctrl-Z unterdrücken.
Um sich das „Ertasten" über die Menüleiste zu ersparen, läßt sich die Datenübertragung über die Funktionstasten steuern. Dazu kann eine Funktionstaste mit der Befehlsfolge belegt werden, so daß ein Druck auf diese Taste den Vorgang auslöst. Aus einem Menüfenster wird die zu übertragende Datei ausgewählt. Das DFÜ-Programm bietet aber auch einen Online-Modus. Mit passendem Modem kann man die Telefonnummer des Teilnehmers wählen lassen.

Star-Planer PC

Als drittes Programm der Star-Serie haben wir noch den Star-Planer getestet. Star-Planer PC ist ein Tabellenkalkulationsprogramm und verfügt über die Fähigkeit, die Daten einer Tabelle grafisch darzustellen. Darüber hinaus lassen sich Tabellen als Datenbanken bearbeiten, denn eine Tabelle läßt sich recht einfach nach Schlüsseln sortieren. Die Tabellenkalkulation von Star-Planer ist kompatibel zu Lotus 1-2-3 in der Version 1A. Lotus 1-2-3 hat sich als Standard der Tabellenkalkulationsprogramme auf dem Markt etablieren können.
Das Programm ist überall dort einsatzbereit, wo Berechnungen anzufertigen sind. Es lassen sich Haushaltskosten abrechnen, Tilgungsraten eines Darlehens berechnen, betriebliche Kalkulationen durchführen, Budgets planen, statistische Auswertungen durchführen oder auch Adressen verwalten.
Nach ihrer Erstellung können die Tabellen grafisch aufbereitet und ausgewertet werden. Star-Planer erlaubt die Übernahme von Daten aus Star-Writer PC oder dBase.
Zum Einsatz des Star-Planers benötigt man einen IBM-PC/XT/AT oder kompatiblen Rechner mit mindestens 256 KByte RAM Arbeitsspeicher und zwei Diskettenlaufwerken sowie einem Monochrom- oder Farbbildschirm. Mit einer gesondert anzufordernden Programmversion läßt es sich auch an PC mit nur einem Laufwerk einsetzen. Zur Darstellung der Grafiken am Bildschirm unterstützt das Programm verschiedene Grafikkarten wie MDA, Hercules-Grafikkarte und für die neuen IBM PS/2 Mo-

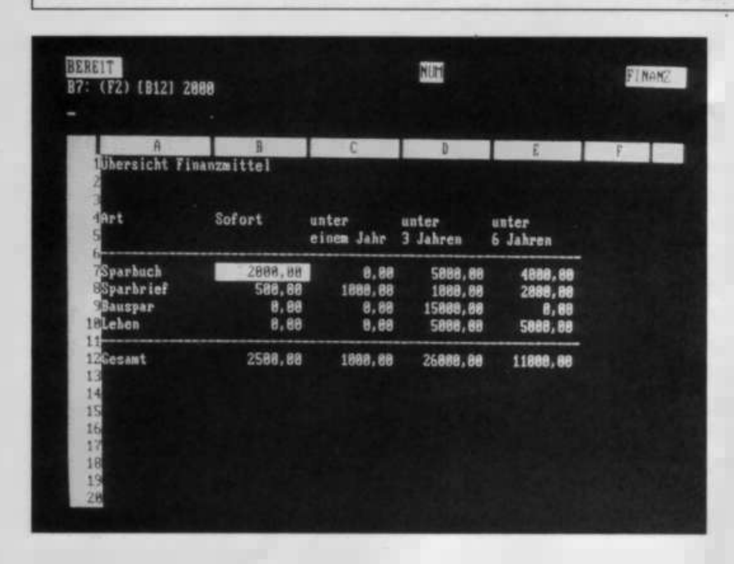


Bild 4. Der Eingabebildschirm ist in numerierte Zeilen und alphabetisch gegliederte Spalten eingeteilt

delle MCGA und VGA. Wird ein Bildschirmadapter verwendet, der mehrere Grafikkarten emuliert, dann wird die EGA-Einstellung benutzt.

Star-Planer PC ist komplett in Maschinesprache geschrieben und aus diesem Grund sehr schnell. Die Daten werden beim Arbeiten nicht auf Diskette ausgelagert, sondern sind stets im Arbeitsspeicher. Deshalb ist die maximale Arbeitsspeichergröße von 640 KByte unter MS-DOS zu beachten. Das Arbeitsblatt von Star-Planer umfaßt maximal 256 Spalten und 2048 Zeilen. Würde man die gesamte Speicherkapazität für ein Arbeitsblatt nutzen, stünden 524288 Felder zur Eingabe bereit. Arbeitsblätter von größerem Umfang müssen in mehrere kleine Arbeitsblätter unterteilt werden. Zum Ausdruck können verschiedene Druckertypen verwendet werden, wie Matrix- und Laser-Drucker. Typenrad-Drucker sind für den Ausdruck von Grafiken grundsätzlich nicht geeignet.

Zum Programm gehört auch eine umfangreiche Hilfefunktion. Sie arbeitet befehlsbezogen und bietet somit an der richtigen Stelle die nötige Auskunft. Dazu wird das Arbeitsblatt vorübergehend gelöscht und der Textbildschirm mit den Bearbeitungshinweisen aufgebaut. Nur der aktuelle Befehl oder der aktuelle Feldinhalt wird angezeigt.

Die Menüzeile wird im Unterschied zu den bereits beschriebenen Star-Programmen nicht ständig am oberen Bildschirmrand geführt. Es wird mit der Taste „Klammer auf" oder dem Backslash eingeblendet. Die Auswahl des Befehles geschieht mit seinem Anfangsbuchstaben oder mit der Positionierung des Cursors und dem RETURN. Je nach der Wahl des Befehls muß ein weiteres Menü aufgebaut werden, um den gewünschten Befehl zu erreichen.

Soll ein Arbeitsblatt gespeichert werden, muß zunächst der Menüpunkt „Datei" ausgewählt und dann der Befehl „Transfer Speichern Ja" bestätigt werden.

Der Eingabebildschirm ist in numerierte Zeilen und alphabetisch gegliederte Spalten eingeteilt (*Bild 4*).

Meist führt die erste Spalte einer Kalkulation Bezeichnungen. Gibt man sie ein, steht die eine oder andere nur noch verstümmelt am Bildschirm. Die Spalten sind standardmäßig nur 10 Zeichen breit. Die Spaltenbreite läßt sich jedoch individuell zwischen 1 und 72 Zeichen einstellen. In manchen Fällen sollte man sich nach der längsten Eingabe richten können, ohne die Zeichenanzahl abzählen zu müssen. Die Breite läßt sich aus diesem Grund mit dem Cursor abtasten.

Zur Justierung der eingegebenen Texte gibt es Labelpräfixe. Spaltenüberschriften lassen sich damit zentrieren oder auch rechtsbündig schreiben. Alle Formate und Justierungen im Arbeitsblatt werden mit abgespeichert. Als Format ist im Star-Planer PC auch die Bestimmung der Nachkommastellen gemeint. Wenn ein Format von zwei Nachkommastellen gewählt wurde, werden die Zahlen nur mit zwei Stellen nach dem Komma angegeben. Das Programm rechnet jedoch mit einer Genauigkeit von maximal 15 Nachkommastellen.

Für die Berechnung komplexer Aufgaben werden Formeln und Funktionen eingesetzt. Einfache Rechnungen lassen sich durch die Positionierung des Cursors auf die entsprechenden Felder und deren Verknüpfung mit dem Operator vornehmen. Jede Formel sollte mit einem Vorzeichen beginnen. Gespeichert wird bei dieser Eingabeweise nicht das Ergebnis, das vordergründig am Bildschirm erscheint, sondern die Berechnung, die zum Ergebnis führt. Die Felder lassen sich absolut, relativ und auch gemischt adressieren. Mehrere Felder können zu Bereichen zusammengefaßt und wiederum in vielen Befehlen und Funktionen verwendet werden. Bereiche lassen sich mit Namen versehen, löschen, kopieren und schützen. Star-Planer bietet mathematische (z. B. LOG, RUNDEN), statistische (z. B. ANZAHL-Argumentliste) Funktionen wie auch Datenbankstatistik-Funktionen (z. B. DANZAHL zur Berechnung der Anzahl aller durch die Kriterienbereiche ausgewählten Werte), Finanzfunktionen (z. B. RATE – berechnet die Rückzahlungsbeträge für ein Darlehen).

Ein Arbeitsblatt kann in Fenster unterteilt und mehrere Arbeitsblätter dürfen miteinander kombiniert werden. Für den Ausdruck wird bei Bedarf ein Bereich der Kalkulation bestimmt. Die Randeinstellung und das Druckformat müssen dazu bestimmt werden. Ist der zu druckende Bereich breiter als die Randeinstellung zuläßt, werden die Spalten, die darüber reichen, auf die nächste Seite ausgegeben. Kopf- und Fußzeilen tragen mit einer Seitenüber- oder auch Seitenunterschrift zur Übersichtlichkeit bei. Genauso läßt sich auch die zur Kalkulation dazugehörende Grafik ausdrucken.

Die grafische Darstellung einer Kalkulation gewährt meist rascher einen Einblick in Statistiken oder Kosten und in deren Tendenzen. Die Informationen, die die Zahlen in den Tabellen liefern, können in einer grafischen Darstellung leichter verglichen und bewertet werden. Deshalb wird an Kalkulationsprogrammen die Forderung gestellt, Grafiken als Auswertung von Tabellen darstellen zu können. Ein Kalkulationsprogramm ohne diese Fähigkeit würde heutzutage gar nicht mehr akzeptiert.

Für die Aussagekraft einer Grafik ist die Leistungsfähigkeit des Programmes, des Computers und seiner Grafikkarte sowie des angeschlossenen Druckers, aber auch in einem besonderen Maße die Auswahl der Darstellungsform ausschlaggebend. Star-Planer PC bietet aus diesem Grund verschiedene Grafiktypen an: Liniengrafik, Balkengrafik, XY-Grafik, gestaffelte Balkengrafik, Kreisgrafik. Jeder Tabelle bzw. jedem ausgewählten Bereich einer Tabelle können mehrere Grafiken zugeordnet werden. Die Grafiken werden ähnlich wie die Bereiche mit Namen benannt und vom Programm verwaltet. Alle Grafiken lassen sich beschriften und mit erläuterndem Text versehen. Überschriften dürfen zwei Zeilen a 39 Zeichen belegen. Werden mehrere Grafikbereiche in einer Darstellung eingebunden, wie beispielsweise in einer Stapelbalkengrafik, dann sind Legenden für die verwendeten Farben oder Schattierungen notwendig. Die Legenden dürfen maximal 19 Zeichen lang sein. Die Datenpunkte in Liniengrafiken und in den anderen Grafikfor-

men kann man mit Datenwerten beschriften. Dazu dienen immer Feldinhalte oder Bereiche. Sie werden von Star-Planer PC übernommen und neben das entsprechende Symbol gesetzt.
Im Star-Planer PC kann man eine Datenbank aufbauen. Die Struktur ihrer Funktionen wird von der Tabelle abgeleitet. Mit den Datenbankfunktionen lassen sich Tabellenbereiche bearbeiten und auswerten. In der ersten Zeile der Datenbank stehen die Feldbezeichnungen, die dazugehörenden Datenfelder sind in Spalten darunter angeordnet. Eine Zeile enthält jeweils einen Datensatz. Ein Datensatz besteht aus mehreren Datenfeldern. Alle Datensätze bilden den Datenbereich. Die Anzahl der Datensätze ist durch die Speicherkapazität begrenzt, da die Datenbank ebenfalls im internen Speicher verwaltet wird. Die Datensätze dürfen bearbeitet, sortiert, gelöscht werden. Datensätze können nach bestimmten Kriterien ausgewählt werden. Mehrere Kriterien lassen sich mit UND und ODER verknüpfen. Relative Vergleiche mit logischen Operatoren sind gleichfalls zulässig. Alle Funktionen des Star-Planer PC werden auch für Datenbanken eingesetzt. Eine Datenbank kann so z. B. statistisch ausgewertet werden. Es lassen sich die Häufigkeitsverteilung ermitteln und Was-Wäre-Wenn-Tabellen berechnen.

Um die Routinearbeiten im Arbeitsblatt zu erleichtern, dürfen einfache Tastaturmakros programmiert werden. Makros werden im Arbeitsblatt unter einem bestimmten Bereichsnamen geschrieben und gespeichert. Um sie in anderen Arbeitsblättern einsetzen zu können, sollte man sie in einem separaten Arbeitsblatt speichern und mit dem Befehl „Transfer Kombinieren" aus dieser Makrobibliothek abrufen. Die einmal erfaßten und gespeicherten Makros sind änderbar. In einem Makro kann ein Schlüsselwort vorgegeben sein, daß sich wie ein Stoppcode auswirkt und eine freie Eingabe abwartet. Erst nach Eingabe des RETURN wird das Makro fortgesetzt. Soll die Spaltenbreite geändert werden, so wird bis zur Eingabe der Spaltenbreite, die nicht immer gleich ist, das Makro abgearbeitet. Zur Eingabe der gewünschten Breite wird das Makro unterbrochen und danach mit RETURN fortgesetzt. In einem Makro kann der Aufruf eines anderen vorgesehen werden, indem der Name eingegeben wird. So lassen sich größere Programme in kleinere aufteilen, die bestimmte abgegrenzte Aufgaben ausführen.

Zusammenfassend...

Alle drei getesteten Programme lassen sich komplikationslos installieren. Die Installationsprogramme führen auch einen ungeübten Anwender über Dialogboxen schnell zum Ziel. Vorausgesetzt wird selbstverständlich die Kenntnis über die zur Verfügung stehende Hardware. Allerdings wird die vorzunehmende Änderung der Datei CONFIG.SYS in der deutschen Dokumentation vorgegeben.

Die drei Programme sind über Menüs zu bedienen. Am oberen Rand des Bildschirmes ist die Menüzeile und unter den einzelnen Menübezeichnungen weitere Untermenüs erreichbar. Ihr einheitliches Konzept findet seine Fortsetzung in der dazugehörenden Dokumentation. Die Handbücher regen zum Arbeiten mit den Programmen an. Dazu trägt ihre einheitliche inhaltliche und äußerliche Gestaltung bei. Es scheint für ihre Erstellung keine englischsprachige Vorlage gegeben zu haben, zumindest sind die Bücher sprachlich davon unbeeinflußt geblieben.

Die eingangs erwähnte Philosophie, die sich hinter den Produkten der Star-Division verbirgt, eine komplette Produktpalette für den deutschen Anwender zu schaffen, wird beim Einsatz der Programme spürbar.
Zum Test stand uns ein AT mit Monochrom-Bildschirm und der NEC Pinwriter P2200 zur Verfügung. □

Termin-Manager

Computer eignen sich bekanntlich vorzüglich dazu, alles und jedes zu verwalten. Das Programm Termin-Manager II von Spiess Computertechnik in München dient zur Terminverwaltung, Fristenüberwachung und Arbeitsplanung. Alle wichtigen Termine und Informationen, die zur Planung und zur täglich Arbeit notwendig sind, werden in Fenstern auf dem Bildschirm bereitgestellt. Am unteren Bildschirmrand sind in einer Menüleiste die Befehle aufgeführt. Sie sind über die zehn Funktionstasten zu aktivieren.
Zur Benutzung des Termin-Manager II benötigt man einen IBM-PC oder dazu kompatiblen Rechner mit mindestens 512 KByte RAM, einem Laufwerk und MS-DOS ab der Version 2.0. Es ist auch auf den neuen IBM-Modellen der Serie PS/2 sowie in einer Netzwerkversion von mehreren Benutzer gleichzeitig einsetzbar. Wir haben den Termin-Manager II für unseren Kurztest auf einem AT mit 512 KByte RAM installiert. Das Programm ließ sich bei der genannten Mindestkonfiguration problemlos installieren. Es benötigt den kompletten Arbeitsspeicher, was den Einsatz von speicherresidenten Programmen verhindert. Mit dem Termin-Manager kann man Tagestermine minutengenau verwalten; gibt keine festen Zeitintervalle. Überschneidungen von Terminen werden vom Programm erkannt und zur Korrektur angeboten. Mehrfach wiederkehrende Termine brauchen nur einmal eingetragen zu werden. Selbstverständlich sind alle Termine änderbar, zu verschieben und auch auszudrucken. Terminen kann man eine Priorität zuordnen.
Wenn der Terminkalender besonders dicht ist, listet der Termin-Manager alle freien Termine auf, die bestimmten Kriterien entsprechen. Zu jedem Termin kann man Notizen – sogar mehrseitige – erfassen. Das Programm bietet eine integrierte Adreßverwaltung. Bei Bedarf lassen sich terminbegleitende Texte – die auch mit anderen Programmen erfaßt sein können – mit ausgewählten Adressen ausdrucken.
Messe- und Seminartermine, Feiertage und Ferien, anstehende Aufgaben, Aktennotizen oder jedes andere zu einem Datum gehörige Merkmal wird in einem speziellen Info-Fenster eingetragen. Zur Auswertung der geleisteten Arbeit kann man sich einen statistischen Überblick geben lassen über z. B. die Summe aller Arbeitsstunden, die für ein bestimmtes Projekt aufgewendet wurden.
Das Handbuch beschreibt die Befehle kurz und knapp. Eine ausführlichere Einführung in die Benutzung der vielfältigen Funktionen wäre gerade für all diejenigen, die vom Tischkalender zum elektronischen Terminplan wechseln wollen, von besonderer Bedeutung. Da eingangs auf die für das gesamte Programm generalisierte Tastenbelegung hingewiesen wird, erwartet man eine tiefergehende Beschreibung der einzelnen Programmteile in den nach ihnen benannten Absätzen. Leider entspricht das Handbuch nicht diesem Verlangen des Benutzers. Es ist weder didaktisch aufbereitet noch sprachlich überarbeitet. Abbildungen, die sich selbst erläutern und einige Zusammenhänge darstellen würden, gibt es nicht. Es bedarf einiger Vorkenntnisse beim Arbeiten am Computer und ein wenig Geduld bei der Erarbeitung der Programmfunktionen. Dann aber kann das Programm die Organisation hilfreich unterstützen.

Dipl.phil. Lore Schönrock

Peter Wollschlaeger

SPC-Modula für den Atari ST

Modula-2 nach Karlsruher Art

SPC-Modula basiert zwar direkt auf dem ETH-Compiler, doch dahinter steckt weit mehr als nur eine weitere Portierung auf den Atari ST. Die Entwickler aus Karlsruhe haben sich einige Besonderheiten einfallen lassen. So nebenbei läßt die Thematik einige wichtige Tendenzen erkennen.

Vor gut zwei Jahren schickte mich mein Arbeitgeber zu Vorlesungen von Professor Wettstein (Universität Karlsruhe) über die Architektur von Betriebssystemen. Was ich dort erlebte, hat mich leicht überrascht. Hier wurden nämlich alle Beispiele in Modula-2 und nicht in C ausgeführt. Die Begründung dafür war für mich auch einleuchtend. Bei so komplexen Programmen, wie es ein Betriebssystem ist, ist die Sprache C nicht sicher genug.

Hier zeichnet sich der erste Trend ab: Modula-2 wird immer wichtiger. Von der Tendenz her ist C die Sprache, die von einigen Software-Häusern favorisiert wird. Modula-2 hingegen wird von den Universitäten bevorzugt und gepusht. Doch die Universitäten entlassen ihre Kinder und Modula-2 dringt dadurch sicher immer stärker auf dem deutschen Markt vor.

Der zweite Trend: Der Atari ST hat sich an den deutschen Universitäten sehr stark etabliert. An immer mehr Instituten kommt der preiswerte Rechner bei Profis zum Einsatz.

Der dritte Trend: Software-Zentrum Karlsruhe. Dieser Trend wird ersichtlich, wenn ich die guten Software-Produkte, die ich im letzten Jahr getestet habe, nach Herstellerorten sortiere. Nach München und Berlin entsteht ein neues großes Softwarezentrum in Deutschland, und das ist Karlsruhe.

Packt man diese Fakten zusammen, sind wir beim Thema, nämlich einem Modula-2-Entwicklungssystem für den Atari ST aus Karlsruhe. „advanced applications Viczene GmbH“ heißt der Hersteller, SPC-Modula (Single Pass Compiler) das Produkt. Für 349 DM wird eine mit 608 KByte gefüllte Diskette geliefert, auf der sich 132 Dateien befinden. Den größten Teil davon findet man in den Ordnern Syslib (64 Dateien) und Stdlib (36 Dateien).

Dazu kommt ein umfangreiches deutsches Handbuch (mit über 300 Seiten) und ein aktiver Update-Service. Das erste (kostenlose) Exemplar der SPC-User-Zeitung ist auch schon eingetroffen.

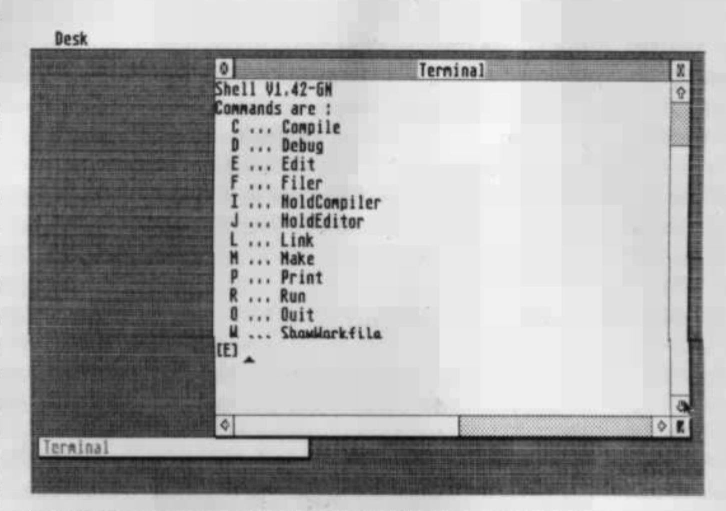


Bild 1. Diese Kommandos kennt die Shell von SPC-Modula

Sinnige Shell

Das Programmier-System wird über eine tastaturorientierte Shell gesteuert, die auf den ersten Blick recht primitiv erscheint. In der Praxis erweist sie sich aber schnell als sehr sinnvoll und gut durchdacht. In *Bild 1* sind die wenigen Kommandos der Shell aufgelistet. Diese Liste erscheint immer, wenn man „H“ für Hilfe eingibt.

Die Shell bietet immer das nächste, logische Kommando als Vorgabe an. Also zuerst „E“, um eine Datei zu editieren. Und wenn man den Editor verläßt, bietet die Shell „C“ für den Start des Compilers an.

Zudem erscheint in der Dateiauswahl-Box immer der passende Name. Dadurch reduziert sich der Weg durch die Turn-Around-Schleife (Editor - Compiler - Linker - Test - Editor...) auf mehrmaliges Drücken der Return-Taste. Übrigens kann sogar die Dateiauswahl-Box (und das zugehörige Return) noch umgangen werden, indem man die Leertaste drückt. Beachten Sie die Kommandos „HoldCompiler“ und „HoldEditor“. Damit werden diese Programme RAM-resident geladen, was sich besonders bei Diskettenbetrieb lohnt. Ansonsten kann ich die Arbeit mit Disketten-Laufwerken nicht empfehlen; eine Festplatte sollte man schon haben.

Einfache Anpassung

Das Print-Utility erzeugt formatierte Ausdrucke und sorgt für eine automatische Hervorhebung der Schlüsselwörter. Geninal einfach ist die Druckerkonfiguration gelöst. Man trägt einfach in das Environment den Namen der Wordplus-Konfigurationsdatei ein und fügt Zeilenlänge und Zeilenzahl hinzu.

Damit wären wir bei der Konfiguration des Software-Paketes angelangt. Das System läuft auf jeder Festplatten-Partition. Man muß nur das ganze Unterverzeichnis „SPC“ von der Diskette in das Wurzelverzeichnis kopieren (kein Kopierschutz, aber Seriennummerverwaltung). Trotzdem kann man über zwei Utilities (Path und SetEnv) eine Konfigurationsdatei anpassen, indem man Pfadnamen vorgibt oder Environment-Variable setzt. Zu diesen Variablen gehören beispielsweise Compiler-Schalter.

Was man damit machen kann, zeigt ein Beispiel: Nehmen wir an, Sie wollen den Editor Tempus alternativ als Editor einsetzen. Dann legen Sie eine Environment-Variable namens „Tempus“ an, tragen dazu den Pfadnamen für Tempus ein (z. B. C:\PASCAL\EDITOR.PRG), und nun kommt's: Sie ändern die Shell! Da diese im Quelltext vorliegt, ist das kein Problem. In *Bild 2* ist die dafür erforderliche Prozedur angegeben (Vergessen Sie die „Imports“ nicht). Nachdem diese Routine der Shell hinzugefügt wurde, schreibe man noch in den Case-Selektor der Hauptschleife

„T: CallTempus“
und man kann nun wahlweise mit „T“ den Editor Tempus aufrufen oder mit „E“ den SPC-Editor wählen.
Die Prozedur ist noch ausbaufähig. So könnte man z. B. noch den Dateinamen und ERR.LST übergeben. Wahrscheinlich lohnt sich das aber nicht.

Modula-Editor

Der mitgelieferte Editor ist window-orientiert und wahlweise über die Tastatur und Pull-Down-Menüs oder über die Maus bedienbar. Das Tempo und die Perfektion von Tempus erreicht er nicht, bietet aber alle erforderlichen Funktionen und eine gute Modula-2-Unterstützung. So kann man Schlüsselwörter kleinschreiben, wobei zwei oder drei Buchstaben reichen. Sobald diese Zeichen für ein Wort signifikant sind, erscheint es komplett in Großschrift. So wird etwa aus „pr“ automatisch „PROCEDURE“. Diese Funktion ist über die Help-Taste ein- und ausschaltbar.
Zusätzlich kann man lange Prozedurnamen oder Strukturelemente als Textmakros unter den 10er-Block legen (die Funktionstasten belegt der Editor). Störend für mich sind das halbseitenweise Scrollen und der Cursor. Dieser steht, als kleines Dreieck, unter der Zeile und ist zumindestens gewöhnungsbedürftig. Ich persönlich fand es am praktischsten, Programme mit dem SPC-Editor einzugeben und Änderungen mit Tempus zu erledigen.

Post-Mortem-Debugger

Den Debugger mit „D“ von der Shell aus aufzurufen, bringt herzlich wenig, denn es handelt sich hier um den ETH-Debugger, der bekanntlich erst nach einem Programmabsturz (post mortem) aktiv wird. Danach kann man in fünf Fenstern, wovon *Bild 3* einige zeigt, auf Source-Ebene das Programm, na sagen wir sezieren, um bei diesen Synonymen zu bleiben. Tatsächlich ist der Debugger sehr nützlich, wenn man sich an ihn gewöhnt hat. Dann geht man sogar soweit, durch simple Anweisungen, wie eine Division durch Null, sozusagen Breakpoints zu setzen. Trotzdem, es gibt bessere Debugger, zum Beispiel den von Megamax-Modula.
Ansonsten gilt hier: Andere Sprachen als Modula-2 haben bessere Debugger, aber die haben es auch nötiger.

Zwei Linker

Normalerweise entwickelt man ein Programm, ohne die Shell zu verlassen. Dabei entsteht kein direkt ausführbares Programm. Die Lösung heißt „Load Time Linking“, wobei die erforderlichen Module erst während des Ladens gebunden werden. Dieses dynamische Laden hat den Nachteil, daß die Module in verschiedenen Dateien liegen und auf unterschiedlichen Pfaden gesucht werden müssen. Da diese Methode das Laden hemmt (daher der Begriff Ladehemmung), gibt es den sogenannten Pre-Linker. Dieser funktioniert im Prinzip wie jeder Linker: er geht von einem Modul aus und sucht sich dann auf den Compiler-Suchpfaden (die Sie hoffentlich alle gesetzt haben) alle Module zusammen, die direkt oder mittelbar importiert werden. Dieser Linker wird über eine Kommando-Datei gesteuert, die neben den Namen des Hauptmoduls und der Zieldatei eine Liste aller Module enthalten sollte, die vom Binden auszuschließen sind (Cut-Library-Prinzip). Sie haben richtig gelesen, der Linker optimiert nicht von allein, Sie müssen es ihm sagen.

Der zweite Linker erzeugt die üblichen Programme, die Sie per Mausklick starten können, jedoch ohne jede Optimierung. Hier hat der Hersteller allerdings schon einen intelligenteren Binder angekündigt. Das scheint neuerdings Mode zu sein, gleiches gilt nämlich auch für Megamax-Modula.
Vorerst jedenfalls erzeugt das Programmchen „Hello World“ gut 100 KByte Code. Nun sagen Sie nicht, daß damit Modula-2 nicht für kleine Desk-Accessories geeignet sei. Die Antwort wird nämlich lauten: Für solche Bagetellen ist Modula-2 auch nicht gedacht.

```
PROCEDURE CallTempus;
VAR TempusPfad: ARRAY[0..59] OF CHAR;
  BEGIN
    IF NOT Environment.Get('TEMPUS',TempusPfad)
     THEN InOut.WriteString('Tempuspfad nicht gefunden');
     RETURN;
    END;
  InOut.Terminate; Terminal.Term; TextWindows.Term;
  IF GemDos.Exec(GemDos.LoadStart,TempusPfad,'','') # 0D
   THEN InOut.WriteString('Ladefehler');
  END;
  TextWindows.Init; Terminal.Init; InOut.Init;
END CallTempus;
```

Bild 2. Der Shell kann man eigene Kommandos hinzufügen

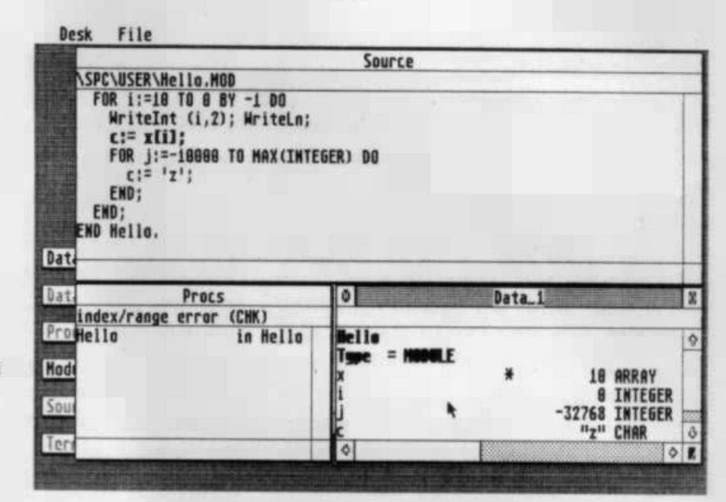


Bild 3. Ein komfortabler Debugger hilft bei der Fehlersuche

Guter Compiler

Der Code, den der Compiler erzeugt, entschädigt allerdings wieder. Erstens geht das mit 5000 Zeilen pro Minute (für Modula-2-Verhältnisse) sehr schnell, und zweitens ist der Code sehr gut optimiert, wie die Benchmarks zeigen. Ich verfüge leider nur über Uralt-Versionen von Megamax- und TDI-Modula-2 (die beide von SPC deutlich geschlagen werden), weshalb ich auf Vergleichszahlen verzichten möchte. *Bild 4* zeigt stattdessen Absolutwerte, die wohl deutlich genug sind.

Co-Routinen ab Version 1.4

Man kann die Shell sehr leicht um neue Funktionen erweitern. In die Version 1.3 des SPC-Modula wurde auf diese Art ein Filer integriert, der in zweierlei Hinsicht bemerkenswert ist. Zum einen ist es ein sehr gutes Programm, das alle Datei-Operationen des Desktop und vieles mehr gestattet, zum zweiten wurde hier schon das Co-Routinen-Konzept angewandt, das offiziell erst mit der Version 1.4 geboten wird.

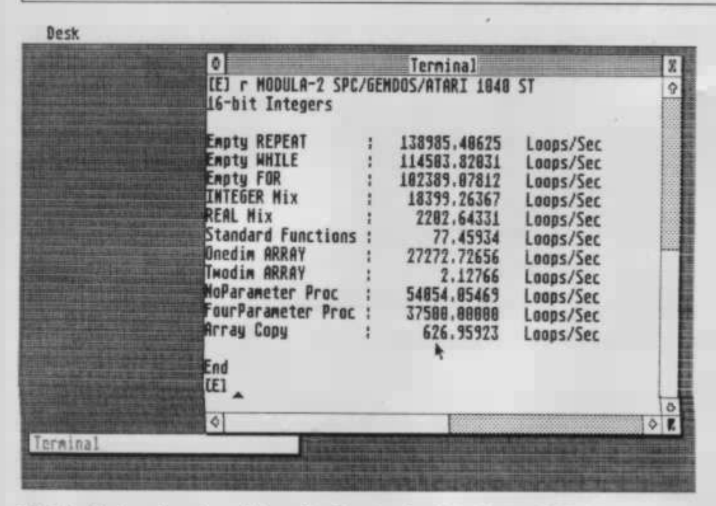


Bild 4. Einige Angaben über die Geschwindigkeit des SPC-Modula unter verschiedenen Bedingungen

```
BEGIN
  SSWiS.Register (Client, 'SSWiSExample', Accept);

  SSWiS.SetMenuTitle (Client, 0,     ' File');
  SSWiS.SetMenuItem  (Client, 0, 0, ' Open');
  SSWiS.SetMenuItem  (Client, 0, 1, ' Close');
  SSWiS.SetMenuItem  (Client, 0, 2, ' -------------');
  SSWiS.SetMenuItem  (Client, 0, 3, ' Quit');

  wxy.X:=  0; wxy.Y:= 300; wwh.X:=  100; wwh.Y:= 2000;
  sxy.X:= 10; sxy.Y:= 100; swh.X:=  200; swh.Y:=  200;
  rwh.X:=  8; rwh.Y:= 16;

  SSWiS.CreateWindow        (Client, 0, Restore);
  SSWiS.PositionWindow      (Client, 0, sxy);
  SSWiS.SizeWindowContent   (Client, 0, swh, swh, swh);
  SSWiS.SetWindowTitle      (Client, 0, 'Window 0');
  SSWiS.PositionWorld       (Client, 0, wxy);
  SSWiS.SizeWorld           (Client, 0, wwh);
  SSWiS.RasterWorld         (Client, 0, rwh);

  SSWiS.PlaceWindowOnTop    (Client, 0);
END Init;
```

Bild 5. Die Programmierung einer grafischen Bedieneroberfläche – einfacher und langsamer als GEM

Schon jetzt funktioniert mit dem Filer folgendes: Durch einen Doppelklick auf einen Dateinamen wird der Editor gestartet. Klickt man nun eine OBM-Datei an, läuft die auch. Die aktuelle Task läuft immer im oben liegenden Fenster. Man kann z. B. einen Text editieren, dabei einen anderen drucken und gleichzeitig auch noch Dateien kopieren. Damit wurde praktisch eine Multitasking-Oberfläche geschaffen, die auf GEM aufsetzt. Damit kann jedes Modul im Multitasking laufen, ohne speziell dafür programmiert sein zu müssen!

Eigenes „GEM"

Man kann mit der VDI/AES-Bibliothek GEM-Programme wie in C üblich schreiben, wird allerdings davon selten Gebrauch machen. SPC liefert das sogenannte SSWiS mit, was für „Small Systems Windowing interface Standard" steht. Damit soll das Dilemma umgangen werden, daß die verschiedenen Computer mit grafischer Benutzeroberfläche zwar in der Funktion das gleiche bieten, aber völlig unterschiedlich zu programmieren sind.

Ob und wann SPC-Modula auf anderen Computern implementiert wird, kann ich nicht absehen, aber auf dem Atari ST läuft das System und bringt dort Vorteile. Fenster, Pull-Down-Menüs und das Event-Handling sind mit SSWiS wesentlich einfacher zu programmieren als unter GEM. *Bild 5* zeigt auszugsweise, wie ein Programm dann aussieht.

Einen Haken hat SSWiS: Die Programme sind langsamer als echte GEM-Programme, was besonders beim Text-Scrollen in Windows auffällt.

Für Profis empfehlenswert

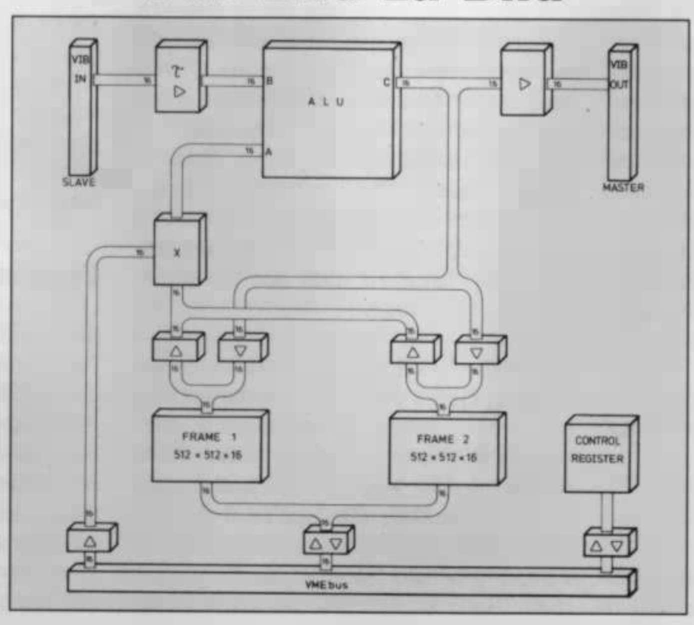
Modula-2 ist eine Sprache für größere Projekte und damit naturgemäß im professionellen Bereich angesiedelt. In diesem Sinne versteht sich auch SPC-Modula, was sich auch in dem recht umfangreichen Handbuch ausdrückt. Hier werden die Tools beschrieben, Besonderheiten wie SSWiS werden nur als gut kommentierte Modula-Source vorgestellt. Für Amateure und Gelegenheits-Programierer ist SPC-Modula nicht gedacht, für Profis ist es empfehlenswert. □

Von Bild zu Bild

Mit dem Bildverarbeitungsprozessor PTP (Picture-to-Picture) von Eltec lassen sich digitale Videobilder in Echtzeit verarbeiten. Der Prozessor verfügt über einen VMEbus-Port für Steuerung und Zugriff auf gespeicherte Bilddaten und zwei VIBus-Ports, über die er Video-Daten (bis 16 Bit) mit bis zu 16 MHz überträgt. Durch einen getrennten Ein- und Ausgang läßt sich der PTP für Pipeline-Verarbeitung einsetzen. Auf dem PTP befinden sich neben einer 16 Bit breiten ALU (Arithmetik Logical Unit) ein Multiplizierer und zwei Bildspeicher mit je 512 KByte.

Das Board kann alle arithmetischen und logischen Bild-zu-Bild-Operationen in Echtzeit bearbeiten. Eine zeitliche Mittelung läßt sich über bis zu 256 Bilder durchführen, das Signal-Rausch-Verhältnis kann dabei über zwei Verfahren verbessert werden: So läßt sich die gewünschte Anzahl von Bildern akkumulieren und nach Normalisierung dem VIBus-Ausgang zur Verfügung stellen oder eine gleitend gewichtete Mittelwertbildung durchführen. Ein Video-D/A-Wandler erlaubt die Darstellung der Ergebnisse auf einem Monitor.

Der VIBus-Standard wurde von Eltec als Rückgrat für eine Pipeline-Verarbeitungs-Struktur von Bilddaten entwickelt, um echtzeitmäßigen Durchsatz unabhängig von der Anzahl der Stufen zu erreichen. □



Die interne Struktur des „Picture-to-Picture"-Prozessors

Stefan Mutschler

Daten im Netz

LAN-Betriebssystem Netware im Detail

Was ein lokales Netz leistet, hängt nicht nur von der LAN-Karte, sondern hauptsächlich vom LAN-Betriebssystem ab. Von Bedeutung sind LAN-Betriebssysteme von Novell, 3COM, Banyan und Microsoft. Einige Eigenschaften von Netware haben wir unter die Lupe genommen.

Netware von Novell ist ein Betriebssystem für Fileserver-orientierte PC-Netze. Bei jedem Netz, das unter Netware läuft, gibt es einen speziellen PC – den Fileserver – der das gesamte System verwaltet. Netware besteht aus zwei Hauptkomponenten – dem Betriebssystem, das auf dem Fileserver läuft, und einer sogenannten Shell, die sich in den einzelnen Arbeitsplatzrechnern um das dort vorhandene PC-Betriebssystem legt (*Bild 1*). Solche Shells gibt es derzeit für DOS 2.x, DOS 3.x. und OS/2. Kürzlich hat Novell auch Netware für Macintosh-Computer angekündigt. Shells haben im wesentlichen die Aufgabe, Befehle der Anwendersoftware an das Betriebssystem, die nicht für den lokalen PC bestimmt sind, abzufangen und an den Fileserver weiterzuleiten. Alles weitere geschieht dort. Das Fileserver-Betriebssystem läuft auf allen IBM-PC/XT/AT und kompatiblen Computern sowie auf allen IBM PS/2-Modellen.

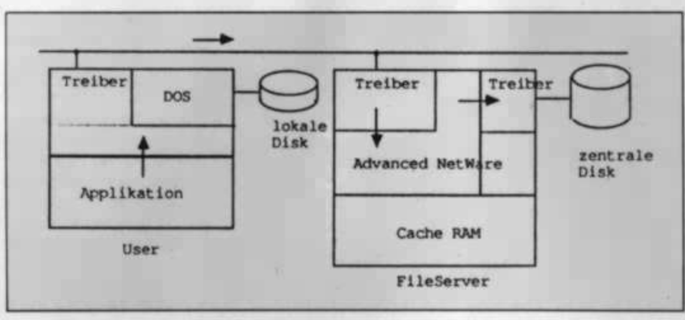


Bild 1. Ein spezieller Treiber an den Arbeitsstationen dient zur Kommunikation mit dem Fileserver

Netware läuft auf der unterschiedlichsten Netzwerk-Hardware: Insgesamt sind es derzeit 43 Systeme, die mit Netware betrieben werden können. Am verbreitetsten sind LANs im Ethernet-, Token-Ring- und Arcnet-Standard (*Bild 2*).

Versionen von Netware

Während man die Shells mit jeder Version von Netware für jedes PC-Betriebssystem konfigurieren kann (per Menüauswahl), muß man beim Fileserver den PC-Typ berücksichtigen und dafür eine spezielle Version erwerben. So gibt es Versionen von Netware, welche speziell den 8088/8086-Prozessor (IBM-PC und kompatible Computer) unterstützen und andere Versionen, welche auf den „protected mode" des 80286-Prozessors (IBM-AT und Kompatible) zugeschnitten sind. Eine spezielle Version für 80386-Maschinen soll es in Kürze geben. Relativ neu sind Versionen für die IBM PS/2-Modelle mit Mikrokanal. Ebenso muß man schon beim Kauf von Netware wissen, ob man den Fileserver „dedicated" (der Server ist ausschließlich Server), oder „non-dedicated" (der Server läßt sich parallel auch als Arbeitsstation benutzen) haben möchte. Für Anwender mit sehr hohen Ansprüchen an die Systemsicherheit gibt es eine fehlertolerante Version, die von Novell SFT Netware (SFT = System Fault Tolerance) genannt wird.

Seit dem vierten Quartal 1987 gibt es eine auf vier Benutzer limitierte Netware-Version (ELS-Netware). Mit dieser Low-Cost-Version soll den Anwendern der Einstieg in die Netzwerktechnik leicht gemacht werden. Für das dritte Quartal 1988 ist eine weitere Low-Cost-Version von Netware angekündigt, bei der das Limit der Benutzer bei acht liegt.

Strukturierungs-Funktionen

Außer bei der SFT-Version von Netware sind die Grundfunktionen des Netzwerk-Betriebssystems im Prinzip gleich. Es muß die Zugriffe aller Netzwerkstationen auf die zentrale(n) Festplatte(n) und Peripherie möglichst schnell und sicher verwalten. Die tragende Rolle im Netzwerk-Theater spielt die Festplatte, wo alle zentralen Programme und Daten gespeichert sind. Festplatten sollten generell gut organisiert und strukturiert sein. Dies gilt für eine Netzwerk-Platte natürlich um so mehr, da hier mehrere Benutzer ihren Datenverkehr abwickeln.

An jeden Fileserver lassen sich eine oder mehrere Platten anschließen. Jede physikalische Disk läßt sich in eine oder mehrere logische Einheiten – sogenannte Volumes – aufteilen, die unter Netware bis zu 256 MByte groß sein können. Jeder Fileserver besitzt mindestens ein solches Volume – das Volume „SYS" – das bei der Installation von Netware automatisch angelegt wird. Der Netzwerkbetreuer (Supervisor) darf weitere Volumes einrichten. Jedes Volume wiederum läßt sich in weitere logische Einheiten aufteilen – die Directories.

Unter Netware kann man für jeden Benutzer 26 logische Laufwerke (A...Z) definieren. In lokalen PC unter DOS sind A und B Diskettenlaufwerke und C bis E lokale Festplatten. Dies kann man unter Netware beibehalten, muß man aber nicht. Auf jeden Fall hat man insgesamt 26 Laufwerkskennungen, die sich beliebigen Directories zuordnen lassen, welche auf bis zu acht verschiedene Fileserver verstreut sein dürfen. Um von einem Directory in ein anderes zu gelangen, gibt man lediglich den entsprechenden Laufwerksbuchstaben ein.

Mit den sogenannten Suchlaufwerken erleichtert man sich die Arbeit im Netz. Sie fassen häufig benötigte Dateien zusammen, auf die man aus jedem beliebigen Verzeichnis oder Unterverzeichnis heraus ohne Angabe eines Directory-Namens zugreifen kann.

Datenschutz im Netz

Bei der Installation von Netware auf dem Fileserver formatiert das Betriebssystem die Festplatte(n) in einem speziellen Format. Selbst wenn man also den Server über eine

Diskette mit DOS booten würde, könnte man die auf den Platten gespeicherten Daten nicht lesen. Das spezielle Aufzeichnungsformat ist die Basis aller weiteren in Netware integrierten Schutzmechanismen. Netware bietet vier Wege, Daten zu schützen:

- Login/Paßwort
- Trustees
- Directory-Schutz
- Datei-Attribute

Diese Schutzmechanismen lassen sich einzeln und in beliebiger Kombination anwenden. Der Zugriff auf die Daten im Fileserver geschieht ausschließlich von den Arbeitsstationen aus. Es gibt zwar zahlreiche Befehle für Operationen an der Fileserverkonsole, jedoch sind diese nur zur Ausführung diverser Kontrollfunktionen und nicht zur Bearbeitung von Daten geeignet. Bevor jemand das Netz benutzt, muß der Netzwerkbetreuer nach der Installation zunächst sogenannte Objekte einrichten. Objekte können Benutzer, Benutzergruppen, weitere Fileserver und einiges mehr sein. Der Netzwerkbetreuer vergibt für jedes Objekt einen individuellen Namen. Wer auf Daten zugreifen möchte, muß sich von einer Arbeitsstation aus beim Fileserver anmelden. Dies geschieht durch das Login-Kommando, mit dem sich der Anwender durch den Login-Namen bei dem gewünschten Server als aktiver Teilnehmer identifiziert.

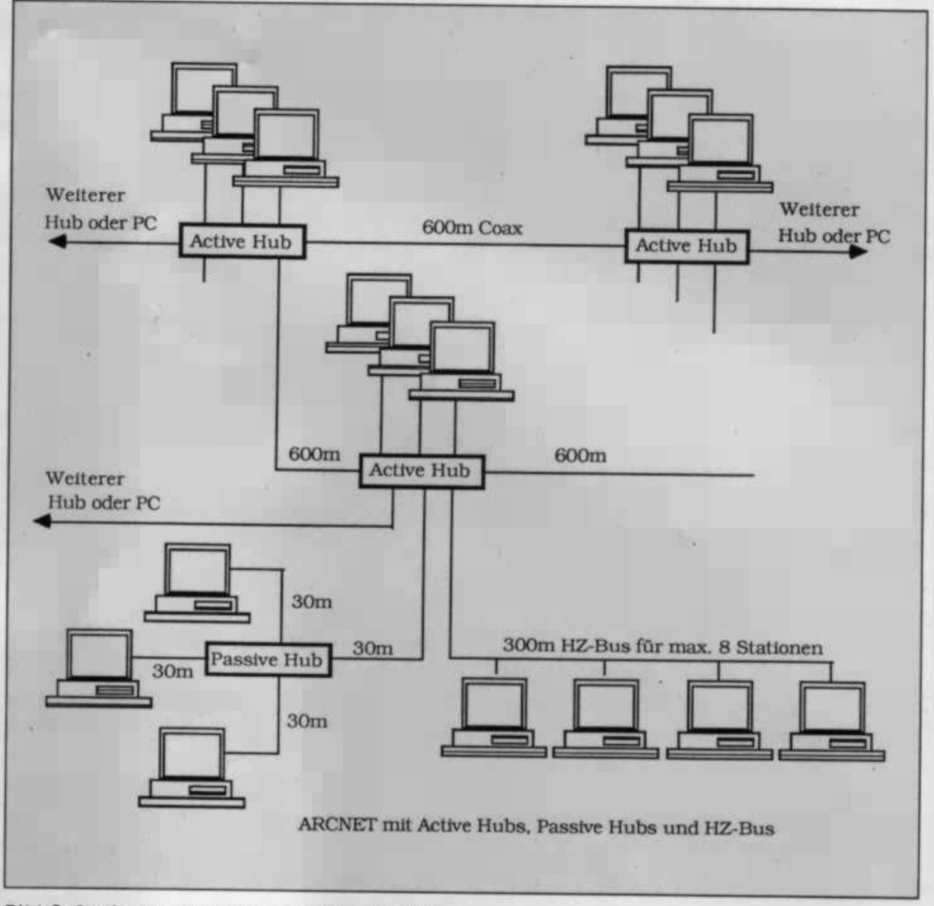


Bild 2. Logischer Aufbau eines Arcnet-LAN

Paßwort-Schutz

Während der Login-Prozedur fragt Netware neben dem Namen grundsätzlich auch nach einem Paßwort. Das Paßwort, das bis zu fünfzehn druckbare Zeichen lang sein darf, kann sich der Benutzer selbst einrichten. Der Supervisor kann Zyklen festlegen, innerhalb derer ein Benutzer sein Paßwort ändern muß. Ist ein Paßwort definiert, muß man es bei jedem Login angeben.

Benutzer-Rechte

Ein dem System bekannter Benutzer kann sich zwar einloggen, aber er hat zunächst noch keinen Zugriff auf Kommandos, Daten und Programme. Der Supervisor muß ihm erst die entsprechenden Rechte (Trustee Assignments) zuteilen.
Mit den „Trustee-Assignments“ legt man für einen Anwender (oder eine Gruppe) die globalen Zugriffsrechte auf das System fest. Diese Rechte gelten prinzipiell für alle Datei-Verzeichnisse, jedoch kann man sie auf beliebiger Directory-Ebene neu definieren. Sie gelten dann für alle darunter liegenden Directory-Ebenen. Ein Benutzer, der einer Benutzergruppe angehört, besitzt automatisch auch deren Rechte.
Der Arbeitsaufwand zur Vergabe von Zugriffsrechten läßt sich mit den „Security Equivalences“ reduzieren. Die Rechte eines Benutzers oder einer Gruppe kann man auf andere Benutzer übertragen. Trägt man zum Beispiel bei einem Benutzer unter Security Equivalences den Namen des Supervisors ein, besitzt dieser alle Rechte des Systembetreuers.
Trustees und Security Equivalences beziehen sich immer auf Benutzer und gelten ab dem gewählten Directory auch für alle Unterverzeichnisse, sofern sie dort nicht neu definiert werden. Wenn man die Zugriffe auf bestimmte Directories einschränken möchte, streicht man die entsprechenden Rechte aus einer Tabelle, in der standardmäßig vom System her prinzipiell alle Rechte eingetragen sind. Diese Tabelle wird „Maximum Rights Mask“ genannt. Die hier spezifizierten Einschränkungen gelten nur für das gewählte Directory, nicht jedoch für Unterverzeichnisse. Sie müssen dort, falls gewünscht, gestrichen werden. Allerdings ist die Maximum Rights Mask benutzerunabhängig – die hier spezifizierten Einschränkungen gelten also immer für alle Netzwerk-Teilnehmer.
Benutzer- und Directory-Rechte lassen sich mit Netware sehr differenziert gestalten. Insgesamt gibt es acht verschiedene Typen von Zugriffsrechten:

R – geöffnete Datei lesen (read)
W – geöffnete Datei schreiben (write)
O – Datei öffnen (open)
C – neue Datei erstellen (create)
D – Dateien löschen (delete)
P – persönliches Directory (parental)
S – im Direcrory suchen (search)
M – Datei-Attribute ändern (modify)

Netware kennt fünf Dateiattribute:

- *read-write/read only:*
  Zugiff zum Lesen erlaubt. Schreiben und Umbenennen sind nicht erlaubt.

- *shareable/non-shareable:*
  Auf non-sharable gesetzte Dateien können nicht von mehreren Benutzern gleichzeitig geöffnet werden. Dies ist wichtig, wenn Anwenderprogramme keine anderen Schutzmechanismen zum Dateizugriff (Record-Locking) besitzen.

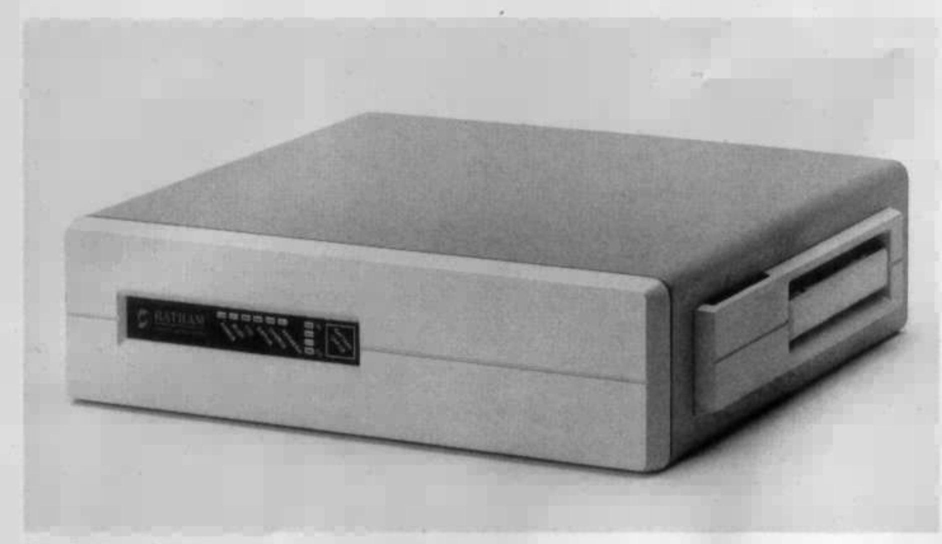


Bild 3. Eine RAM-Disk für Netware-LANs

– *execute only:*
Die Datei muß eine Programmdatei sein. Die einzig mögliche Operation ist der Start des Programms.

– *hidden:*
Wie unter MS-DOS ist eine verborgene Datei im Directory nicht sichtbar.

– *system:*
Es handelt sich um eine Systemdatei.

LAN-Tuning

Es gibt eine Reihe von Maßnahmen, ein Computer-Netz durch ein Betriebssystem schneller zu machen. Ein gutes Betriebssystem verringert deutlich die Schwächen der anderen Netzkomponenten. Die Festplatte ist wegen der Masseträgheit des Schreib-/Lese-Kopfes und wegen der begrenzten Drehzahl der Plattenstapel in den meisten Fällen der größte Engpaß eines Netzes.
Die wirksamsten Maßnahmen sind daher jene, welche die mechanische Trägheit des Schreib-Lesekopfes der Festplatte kompensieren. Bei jedem regulären Festplattenzugriff fährt der Schreib-/Lese-Kopf zuerst auf die Spur mit dem Inhaltsverzeichnis und dann erst auf die eigentliche Datenspur.
Mit dem „Directory Caching" umgeht man teilweise die Trägheit der Mechanik. Hierbei wird das gesamte Inhaltsverzeichnis der Festplatte zeitweilig in den Arbeitsspeicher des File-Servers übertragen. Alle weiteren Zugriffe erübrigen das vorherige Laden des Dateiverzeichnisses. Von Zeit zu Zeit wird der Directory-Inhalt auf die Festplatte kopiert, aber nur dann, wenn kein Festplattenzugriff stattfindet und wenn der Directory-Inhalt durch Hinzufügen neuer Daten oder Programme, beziehungsweise durch einen Löschvorgang, geändert wurde. Da für jeden Zugriff nur noch eine statt zwei Positionierungen nötig sind, halbiert sich durch das Directory Cashing die effektive Zugriffszeit.
Das „Hashing-Verfahren" wandelt jeden Directory-Eintrag in einen binären Wert um und speichert diese Einträge sortiert nach der Größe im RAM-Directory ab. Gesucht wird nun binär, statt wie unter DOS sequentiell. Unabhängig von der Größe des Verzeichnisses wird der gewünschte Eintrag in maximal fünf Suchschritten gefunden. Je größer die Zahl der Dateieinträge im Directory, umso mehr bringt die Hashing-Technik Geschwindigkeitsvorteile.
Im Netz arbeiten oft Rechner mit bestimmten Dateien häufiger als mit anderen. Dies können beispielsweise Kunden-, Lieferanten- oder Artikeldateien sein, auf die regelmäßig immer wieder zugegriffen wird. Das Laden solcher Dateien in den Hauptspeicher dauert schon bei Einplatz-PC oft mehrere Sekunden. Hier helfen RAM-Disks weiter. Mittels eines speziellen Programmes wird ein Teil des freien Hauptspeichers als Laufwerk definiert, welches anschließend wie eine Diskettenstation – nur eben um ein Vielfaches schneller – angesprochen werden kann. Allerdings muß man vor Beginn der Arbeit die gewünschten Dateien in die RAM-Disk laden.
In einem Netz, wo beispielsweise zehn Stationen simultan auf die zentrale Platte zugreifen wollen, addieren sich natürlich für den Zehnten die Zugriffszeiten aller Vorgänger. Hier bemerkt der Anwender Wartezeiten, die sowohl unter rationellen, als auch unter ergonomischen Gesichtspunkten untragbar sind.
Mit „File Caching" kann man die Wartezeiten drastisch verringern. Bei diesem Verfahren kopiert der Rechner automatisch die häufig benutzten Speicherbereiche der Festplatte in den Hauptspeicher. File Caching ist mit einer RAM-Disk vergleichbar. Jedoch muß sich hier der Anwender nicht um die Verwaltung der Einträge kümmern, weil der Fileserver den Inhalt des Cash-Speichers jeder Änderung der Zugriffshäufigkeit auf die Dateien anpaßt. Lesen, Ändern oder Schreiben innerhalb dieses RAM-Bereiches erfolgt etwa einhundertmal schneller als auf der Festplatte. Auch hierbei werden – wie beim Directory-Cashing – die RAM-Inhalte in den Arbeitspausen auf die Festplatte zurückgeschrieben, um bei einem etwaigen technischen Defekt oder bei Stromausfall einen größeren Datenverlust zu verhindern.
Schneller wird der Zugriff beim File Caching tatsächlich nur auf häufig benutzte Plattenbereiche. Die Zugriffsgeschwindigkeit für andere Bereiche bleibt von der Festplattenzugriffszeit abhängig. Außerdem hängt der Geschwindigkeitszuwachs natürlich sehr stark davon ab, wieviel freien RAM das Betriebssysstem im File-Server findet. Für alle diese Verfahren der RAM-Nutzung sollte möglichst viel Speicher bereitgestellt werden. Für ein Netz mit etwa 15 Stationen sind drei bis vier MByte empfehlenswert. Das Maximum, beschränkt durch den Adreß-Bus des 80286-Prozessors, liegt bei 16 MByte.

Es ist sehr wahrscheinlich, daß die Datenbereiche, welche die verschiedenen Stationen nutzen, völlig frei über die Festplattenoberfläche verteilt sind. Und es ist ebenso wahrscheinlich, daß die Bereiche in chaotischer Folge aufgerufen werden. Das Resultat ist, daß der Schreib-Lesekopf der Festplatte, welcher die Zugriffsaufträge der Reihe nach abarbeitet, über die Hunderte von Spuren hin- und herspringen muß. Bei dieser sequentiellen Arbeitsweise wird viel Zeit vergeudet. Effektiver ist es, wenn dicht aufeinanderfolgende Zugriffsbefehle zunächst zwischengespeichert und dann so umsortiert werden, daß der Kopf mit einer Bewegung von außen nach innen nacheinander alle auf seinem Weg liegende Positionen anspricht, ohne hin- und herspringen zu müssen. Innen angelangt, werden die Zugriffswünsche für den Rückweg nach außen umsortiert. Das klingt umständlich – tatsächlich ist die Wegeoptimierung erheblich schneller, als der „dumme" Suchlauf und schont ganz nebenbei auch noch die Festplatte. Dieses „Elevator-Seeking"-Verfahren (elevator: Fahrstuhl) bringt eine beträchtliche Verringerung der Suchzeiten, was sich umso mehr bemerkbar macht, je stärker das PC-Netz beansprucht wird.

Mit zusätzlicher Hardware läßt sich ein als Fileserver eingesetzter Computer beschleunigen. Diese Komponenten – etwa Disk-Coprozessor-Boards (DCBs) für eigenständiges Disk-I/O, oder DOS-analoge RAM-Disks (erhältlich bis 80 MByte) können bei Bedarf nachgerüstet werden. So ist z. B. Batram (*Bild 3*) eine RAM-Disk für Netware-LANs. Der Speicherkoffer wird über einen eigenen Controller an den Server angeschlossen. Er läßt sich wie eine Festplatte ansprechen – nur eben erheblich schneller (Faktor 10...30).

Eine weitere Steigerung der Geschwindigkeit erreicht man, wenn möglichst viel Rechenleistung auf den Netzwerk-Interfacekarten selbst zur Verfügung steht. Je intelligenter die Karte ist, umso mehr kann sich der Fileserver um andere Dinge wie Cache-Verwaltung oder Umordnung der Reihenfolge für die Festplattenzugriffe kümmern.

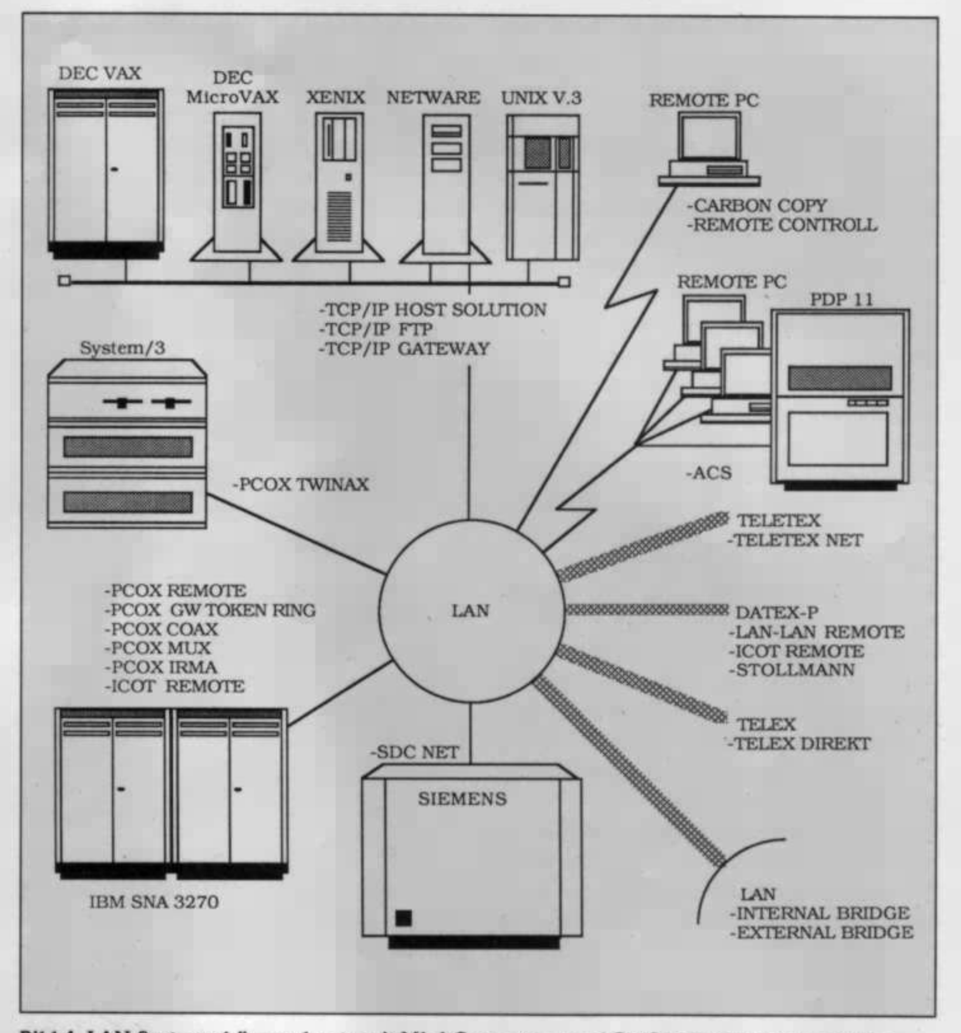


Bild 4. LAN-Systeme können heute mit Mini-Computern und Großrechnern sowie mit den meisten Postdiensten verbunden werden

Sicherheit im Netz

Hier wird ein Punkt angesprochen, dessen Bedeutung einem Netzwerkanwender oft erst dann klar wird, wenn einmal etwas passiert ist. Man kann sich leicht den entstehenden Verlust an Zeit und Geld vorstellen, wenn ein ganzes Netz für einige Stunden stillsteht. In einem Netzwerk sind die Anforderungen an die Ausfallsicherheit weit höher, als in einem Einplatz-System. Andererseits steigt durch die Zahl der elektrischen Komponenten auch die Wahrscheinlichkeit eines Fehlers. Für Netware-Netze gibt es daher eine Reihe von Hilfsmitteln, die den Betrieb sichern. Zu einer soliden LAN-Anwendung gehört ein Backup-Laufwerk (mit Backup-Software, die zum Netzwerk-Betriebssystem kompatibel ist) und mindestens für den Fileserver eine unterbrechungsfreie Stromversorgung. Novell bietet bei Netware eine Reihe von Sicherheitsmechanismen für die Server-Platte, beziehungsweise für den kompletten Server an.

Wie in jedem System mit Festplatte werden auch bei einem File-Server die Dateiverzeichnisse am meisten angesprochen. Hier wird jeder Neueintrag von Daten, jede Dateiverlängerung und -Verkürzung und jede Löschung registriert. Natürlich nutzt sich dieser Datenbereich nicht stärker ab als jeder andere auf der Platte, dennoch steigt die Gefahr eines Fehlers linear mit der Nutzungshäufigkeit an. Kann das System ein Directory nicht mehr lesen, ist der komplette Festplatten-Inhalt nicht mehr ansprechbar und somit verloren. Daher verwaltet Netware zur Erhöhung der Sicherheit die Dateiverzeichnisse und Disk-Tabellen doppelt.

Ein weiterer Mechanismus zur Erhöhung der Datensicherheit stellt die Bit-Fehler-Erkennung und -Korrektur dar. Netware verwendet ein spezielles Aufzeichnungsformat für die Festplatte, das eine sichere Fehlerkorrektur bei bis zu 11-Bit-Fehlern pro Sektor erlaubt.

Eingebaute Fehlertoleranz

Netware SFT I (SFT = System Fault Tolerance) reserviert bei der Installation des Netzes bestimmte Zylinder der Festplatte. Wenn während des Betriebes auf einzelnen Sektoren der Festplattenoberfläche Fehler auftreten, werden die Daten des entsprechenden Sektors in die vorher reservierten Spuren übertragen und der fehlerhafte Sektor wird in die Bad-Block-Tabelle aufgenommen. Solche Fehler werden von Netware automatisch „korrigiert". Novell nennt diesen Mechanismus „Hot Fix".

Um überhaupt feststellen zu können, daß ein Fehler auf der Platte aufgetreten ist, bedient sich SFT I eines Verfahrens, welches „read after write verification" (RAWV) genannt wird. Nach jedem Schreibzugriff werden die Daten sofort gelesen und mit der noch im Speicher vorhandenen Originalinformation verglichen. Tritt hier ein Fehler auf, kommt „Hot Fix" zum Einsatz. „Hot Fix"/RAWV vermeidet so, daß ein komplettes Netz still steht, nur weil ein lapidarer Lesefehler aufgetreten ist. Damit vermeidet SFT1 nicht nur das Neuformatieren der Platte und das Einlesen eines Sicherungsbandes (Backup-Kassette), sondern der Netzbetrieb geht schlicht ohne Panne weiter. Die SFT-Stufe I ist ab Version 2.1 Bestandteil jeder Netware-Version.

Gespiegelte Festplatten-Systeme

Der Ausfall einer Station ist für den restlichen Netzwerkablauf nicht störend – der jeweilige PC, XT, AT oder PS/2 wird einfach ausgetauscht. Kritischer ist allerdings der Ausfall der Festplatte oder auch des

Disk-Controllers im Fileserver. Beide sollten in einem Netz mit erhöhten Sicherheitsanforderungen doppelt vorhanden sein.
SFT II kann wesentlich mehr als SFT I. So spiegelt es Festplatten oder auch, zusammen mit externen Festplatten, komplette Plattensysteme. Spiegeln heißt in diesem Fall, daß der logische Inhalt einer Festplatte auf einer zweiten Platte noch einmal abgespeichert ist. Damit kann jeder Benutzer des Netzes ohne Störung weiterarbeiten, auch wenn, je nach Ausbaustufe, Festplattenlaufwerke, oder sogar gesamte Disksysteme ausfallen.
Das fehlertolerante Betriebssystem schreibt alle Daten zuerst auf die eine, dann nochmals auf die andere Festplatte, und überprüft natürlich durch „read after write", ob beide Festplattendaten lesbar sind. Auf die Ersatzplatte (slave disk) schaltet das Betriebssystem erst im Falle eines nicht reparierbaren Fehlers um. Dann zeigt es dem Benutzer des Netzes an, daß nur noch eine der beiden Festplatten in Betrieb ist und bei nächster Gelegenheit die defekte Festplatte ausgetauscht werden sollte.
Zwar sind heutzutage Festplattenfehler relativ unwahrscheinlich (natürlich stark von der Qualität der Platte abhängig), bei einer kritischen Anwendung, wie beispielsweise der Auftragsabwicklung eines Betriebes, darf jedoch auch ein seltener Fehler nicht zur Unterbrechung des Betriebsablaufes führen.

Plattenfehler können nicht nur von der Platte selbst verursacht verden, sondern auch von dem Controller, an dem sie angeschlossen sind. Bei den Controllern gibt es, genau wie bei den Platten, erhebliche Qualitätsunterschiede. Würden zwei gespiegelte Platten an ein und demselben Controller angeschlossen sein, würden im Falle eines Controller-Fehlers beide Platten ausfallen und die doppelte Datenführung wäre somit sinnlos. Aus diesem Grund kann man unter Netware SFT II die gespiegelten Platten mit zwei getrennten Controllern verbinden. Neben dem Vorteil der höheren Sicherheit steigt als Nebeneffekt auch die Geschwindigkeit des Netzes nochmals, da zwei Prozessoren das Prüflesen beider Festplatten nebeneinander und nicht nacheinander durchführen.

Transaction Tracking

SFT-Stufe II hat noch weitere Eigenschaften zum Sichern des Netzbetriebes. Eine davon ist UPS-Monitoring, welche im Falle eines Stromausfalles (bei angeschlossener Notstromversorgung) Warnungen an die Benutzer ausgibt und bei Erschöpfung des Energievorrates der Notstromversorgung Dateien schließt und den Server geordnet herunterfährt. Die zweite wichtige Eigenschaft wird Transaction Tracking System, auch kurz TTS genannt. Unter einer Transaktion (engl.: transaction) versteht man eine Folge logisch zusammengehörender Aktionen, die Operation auf die gemeinsam gespeicherten Daten ausführen [1].

Zitat des Monats

Statistisches RAM

Ein Schreib-/Lesespeicher, bei dem die Speicherschaltkreise ihren Inhalt behalten, solange sie unter Spannung stehen. Die Abkürzung RAM steht für „Random Access Memory" – also für Schreib- und Lesespeicher.

aus: IC-Wissen 3/88, Seite 40

In einem komplexen Datenbank-System werden zum Beispiel bei Buchungs-Vorgängen mehrere Dateien geändert. Beim Bearbeiten einer Rechnung wird automatisch die Zahl der gelieferten Menge eines Artikels vom Lagerbestand abgezogen, der Betrag wird als Kundenumsatz der Kundendatei zugeordnet, zur Datei „offene Posten" addiert und bei den Vertreter-Provisionen vermerkt. Das Aktualisieren all dieser Dateien – eine Folge von Operationen – wird als Transaktion bezeichnet. Da der Computer eine Transaktion nicht gleichzeitig, sondern nur nacheinander bearbeiten kann, vergeht hier – in den Dimensionen eines Computers – eine verhältnismäßig lange Zeitspanne. Fällt während einer Transaktion eine wichtige Server-Komponente aus, ist ein Teil der Daten bereits geändert, ein anderer Teil aber noch nicht. Der Datenbestand ist also nicht mehr integer.
Eine Behebung dieses Fehlers ist sehr aufwendig.
Vereinfacht dargestellt läßt sich TTS wie folgt beschreiben: Das Anwenderprogramm teilt dem Netzwerk-Betriebssystem den Beginn einer Transaktion mit. Ab jetzt werden alle sequentiellen Veränderungen nicht mehr in den aktuellen Dateien durchgeführt, sondern an getrennter Stelle – einem naheliegenden freien Bereich auf der Festplatte. Erst wenn das Anwenderprogramm signalisiert, daß die Transaktion erfolgreich zu Ende gebracht wurde, werden alle Verweiszeiger des Datenbank-Systems so umgelenkt, daß ab sofort der geänderte Bereich als aktueller Datenbestand gilt. Die Umlenkung der Verweiszeiger erfolgt gleichzeitig und ist somit nicht unterbrechbar. Es sind also immer entweder alle Dateien geändert, oder keine. Professionelle Datenbank-Systeme, wie etwa das Programm ZIM, sind heute in der Regel mit der TTS-Funktion gerüstet. ZIM speziell unterstützt TTS sowohl mit Hilfe von Netware, als auch völlig selbstständig. SFT II ist in allen SFT-Netware-Versionen (ab Release 2.1) enthalten.

SFT III: Gespiegelte Fileserver

Schadhafte Stellen auf der Festplatte repariert SFT I, um defekte Platten und Controller kümmert sich SFT II – bleibt als letzte Komponente, die den Netzwerk-Betrieb empfindlich stören könnte, noch der Fileserver selbst. Sehr häufig – besonders bei Billig-ATs – sind System-Abstürze durch einen Parity-Fehler im RAM-Speicher. Um gegen jeden Fehler im Fileserver gewappnet zu sein, entwickelt Novell gerade SFT III. Laut adcomp, Distributor für LAN- und Kommunikationssysteme, wird SFT III jedoch erst frühestens Ende dieses Jahres zur Verfügung stehen. Bei SFT III sollen nicht nur die Platten, sondern tatsächlich getrennte Fileserver im „Spiegelungs-Verfahren" arbeiten. Eine Software, welche permanent alle Dateien vom Arbeits-Server auf einen Backup-Server kopiert, ist bereits bei adcomp erhältlich. Im Notfall erlaubt „LAN-Shadow" das Umschalten auf den redundanten Server.
Ein professionelles Netzwerk-Betriebssystem umfaßt also eine Menge Aufgaben. Hinzu kommen noch Kommando-Sets für die Benutzer und den Netzwerkbetreuer, Druckersteuerung und externe Kommunikation.
Durch die LAN-seitig gebotenen Kommunikationsfähigkeiten lassen sich heute heterogene Netze mit nahezu beliebiger Konstellation realisieren (*Bild 4*). Die Verfügbarkeit professioneller Standard-Betriebssysteme für Netze ist im wesentlichen für den bisher erlebten und weiter allseits prognostizierten Erfolg von LAN-Systemen verantwortlich.

Literatur

[1] *Hans Robert Hansen*: Wirtschaftsinformatik I. Fischer Verlag, Stuttgart.

[2] *Stefan Mutschler*: Lokale PC-Netzwerke und Kommunikation. Vogel-Verlag, Reihe Chip Special, ISBN 3-8023-0957-X.

Helga Schmidt

Die (un)entbehrlichen Helfer?

Scanner erobern sich einen festen Platz im Büro

Zu einem professionellen Desktop-Publishing-System gehört ein Scanner wie selbstverständlich dazu. Mit Hilfe des Scanners lassen sich Fotos, Grafiken, Firmenlogos und auch Texte in den Computer einlesen, dann weiterverarbeiten oder in bereits bestehende Dokumente integrieren.

Dem steigenden Bedarf entsprechend sind im letzten Jahr zahlreiche neue Scanner vorgestellt worden. Bei Preisen von 600 Mark bis über 200.000 Mark ist, abgesehen von den Anforderungen an die Qualität, für fast jeden Geldbeutel etwas dabei.

Scanner erinnern in ihrer Technik ein wenig an Fotokopierer, nur erstellen sie eben keine Kopie auf Papier, sondern digitalisieren eine Vorlage und übertragen die Daten an den Computer. Eine Lichtquelle wird an dem Foto oder der Abbildung vorbeigeführt und tastet die Vorlage ab. Die hellen Teile eines Fotos reflektieren das Licht stärker, während die schwarzen Bereiche das Licht absorbieren. Die Graustufen entstehen durch den jeweiligen Schwarz-Weiß-Anteil der verschiedenen Bereiche. So ergibt sich aus den erfaßten Mustern eine digitalisierte Form der ursprünglichen Vorlage. Zur Erfassung des reflektierten Lichts setzen fast alle Scanner den sog. CCD (Charged Coupled Device) ein, der Licht in analoge Signale umsetzt. Meist werden nicht alle Grauwerte zu 100 Prozent erfaßt. Wie hoch der Prozentsatz ist, richtet sich nach der Auflösung, die der Scanner bietet bzw. der Anzahl der Graustufen. Diese richtet sich nach der „Bit-Tiefe", also danach, wieviel Bit für die Speicherung eines Pixels zur Verfügung stehen. Bei einer Tiefe von 4 Bit sind das 16 Graustufen, bei einer Bit-Tiefe von 6 können bereits 64 Graustufen erfaßt werden.

Die Qualität eines Scanners läßt sich u. a. dadurch feststellen, daß man zum Testen eine schräge Linie oder eine Freihandzeichnung statt eines Fotos benutzt. Bei Kreisen und Linien sind sog. „Treppenstufen" viel schneller zu erkennen als bei einem gescannten Foto.

Zwei verschiedene Techniken beim Abtasten einer Vorlage werden angewendet: die Flachbett- und die sog. Rollen-Technologie. Beim Flachbett-Scanner werden, ähnlich wie beim Kopierer, die Vorlagen auf eine Glasplatte gelegt. Die Lichtquelle wird unter dem Dokument bewegt und erfaßt die Vorlage, indem sie systematisch jede Zeile abtastet. Der Rollen-Scanner funktioniert wie ein Einzelblatt-Einzug beim Kopierer. Jede Vorlage wird über den Sensor gezogen. Rein von der technischen Qualität macht es keinen Unterschied, ob eine Vorlage mit 300 dpi per Flachbett oder per Rolle erfaßt wird. Doch beim Rollenscanner kann es schon einmal passieren, daß das Blatt nicht ganz gerade eingezogen wird, vor allem, wenn es sich um kleinere Formate handelt. Sollten Sie z. B. ein Paßbild erfassen wollen, weil Sie Ihr Konterfei auf Ihrem Briefbogen plazieren möchten, wird dies nicht funktionieren. Der Flachbett-Scanner dagegen macht da keine Schwierigkeiten. Sie legen das Bild auf die Glasplatte – und schon kann's losgehen. Mit Hilfe der Flachbett-Technik lassen sich auch Vorlagen aus Zeitschriften oder Büchern mühelos verarbeiten. Das ist beim Rollen-Scanner nicht möglich, es sei denn, Sie lösen die jeweilige Publikation in ihre Einzelteile auf. Auch bei stabilen Vorlagen wie z. B. Karton streikt er.

Es ist bei Laserdruckern und Laserbelichtern nicht möglich, die Größe der einzelnen Punkte zu variieren. Die Darstellung von Grauabstufungen muß daher über den „Dither"-Prozeß erfolgen. Eine Gruppe von Pixeln wird zur Erzeugung eines Grauwertes zusammengefaßt und die Anzahl der innerhalb einer festgelegten Fläche auf schwarz gesetzten Pixel bestimmt dann deren Grauwert. Für 64 Graustufen benötigt der Scanner eine Fläche von mindestens 8x8 Pixel.

Zwar geben die meisten Hersteller an, eine Speichererweiterung sei für ihren Scanner nicht erforderlich, doch stimmt das nur rein theoretisch. Eine gescannte Seite in DIN A4 beansprucht ungefähr 1 MByte Speicherplatz. Wenn Sie für umfangreichere Dokumente mehrere Abbildungen neben dem Text unterbringen wollen, wird es auf einer 20-MByte-Festplatte sehr schnell eng. Werden die Bilder darüber hinaus noch editiert, z. B. vergrößert, wird zusätzliche Speicherkapazität beansprucht. Für monochrome Abbildungen in A4 bei einer Auflösung von 300 dpi sollten im RAM ca. 2 MByte zur Verfügung stehen, für Farb-Abbildungen sollten Sie ca. 4 MByte Speicherplatz einkalkulieren.

Neben der Erfassung von Abbildungen und Fotos rückt ein weiteres Einsatzgebiet von Scannern immer mehr in den Vordergrund – die Texterfassung! Was vor gut einem Jahr fast noch wie Zukunftsmusik klang, ist inzwischen Realität. OCR-Software macht es möglich. OCR bedeutet „Optical Character Reading", also optische Zeichenerkennung. Wie bei den grafischen Vorlagen, wird auch eine Textseite als Pixelmuster über den Scanner in den Arbeitsspeicher geladen. Dort werden die einzelnen Buchstaben sequentiell mit den Zeichen-Bitmustern der verschiedenen Schrift-Fonts verglichen. Stimmt die Vorlage mit dem Vergleichsmuster überein, wird dem jeweiligen Muster ein ASCII-Wert zugeordnet.

Ist der Text fehlerfrei von der Software „verstanden" worden, läßt er sich mit einem Textverarbeitungs- oder DTP-Programm editieren und verarbeiten. Komfortable OCR-Software vergleicht parallel zur Buchstabenerkennung den Text mit einem Lexikon, wodurch die Fehlerquote auf ein Minimum reduziert wird.

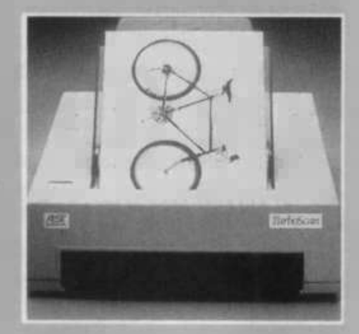
Fragen Sie beim Kauf, welche Schriftarten das OCR-Programm verkraftet, und mit welchem Textprogramm die erfaßten Texte weiterbearbeitet werden können. Es hilft nichts, wenn Sie mit Word arbeiten, das Programm aber Wordstar-Format liefert.

In unserer Marktübersicht haben wir versucht, alle auf dem deutschen Markt angebotenen Scanner zu erfassen – es ist uns fast gelungen. Das Spektrum reicht dabei vom „Handy-Scanner" bis zum Hochleistungsgerät für große Lithoanstalten. Alle Daten beruhen auf den Angaben der Hersteller bzw. Händler; für deren Richtigkeit übernehmen wir keine Gewähr. Auf Anfrage nennen wir Ihnen gern die Anschriften der Anbieter (bitte evtl. die mc-Kontaktkarte ausfüllen).

Scanner im Vergleich

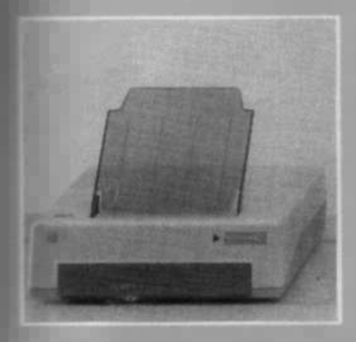






Anbieter/ Hersteller	Afga/Agfa	Agfa/Agfa	AST Research/AST
Name/ Scannertyp	S 600 GS/Flachbett	Focus 800/Flachbett	AST Turboscan/Rollen
Preis inkl. MwSt.	12 642,–/11 566,–	13 211,–	4292,–
Lieferumfang	PC-SW/Apple-SW	Mc-View/Mc-Scan-Software	k. A.
Technologie/Sensor	CCD	CCD	CCD
Schnittstellen	SCSI	SCSI	seriell
Speicherkapazität min./max.	4 MByte/16 MByte opt.	4 MByte/4 MByte	k. A.
Abtastgeschwindigkeit 1 S. A4	4 s bei 75 dpi	8 s bei 100 dpi	11 s
Übertragung byteweise/kompr.	nein (SCSI)	k. A./k. A.	ja/ja
Arbeitsmodi	Linien, Dither, Halbton	Linien, Halbton	Linien, Halbton
Vorlagengröße max./min.	A4/frei wählbar	A4	212 × 350 mm/76 × 76 cm
Auflösung in dpi (Stufen)	75–600 in 13 Stufen	100–800	200, 300
Scanfläche einstellbar/Formate	ja/max. A4	ja/A4	ja/k. A.
Grauwerte/Wechsel möglich	64/ja	64/nein	33/k. A.
Farbe/Blindfarben	nein/keine	nein/k. A.	nein/k. A.
Bildschirmanzeige der gescannten Vorlage	ja	ja	ja
Ausschnitte festlegen/wie?	ja/Maus	ja/k. A.	ja/k. A.
Ausschnitte getrennt bearbeiten?/speichern	ja/ja	ja	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/bis 800 % ja/bis 12 %	ja/bis 800 % ja/bis 12 %	ja/k. A. ja/k. A.
Software-Voraussetzungen	Windows, 2.0 bei IBM	Windows	Windows
Grafikdateiformate	TIFF, GEM, PC Paint Brush, EPSF, PC Paint, Illustrator	TIFF, GEM, PC Paint, PC Paint Brush, IMG	TIFF, GEM, PC Paint, PC Paint Brush, ASCII, PIC, Windows, Eyestar
OCR-fähig/Zahl der Schriften	in Vorb. f. IBM PC	nein	k. A.
Graf. Editierfunktionen	alle üblichen, Vektorisieren, Rotation, Entrastern, Kantenverschärfung	alle üblichen, Vektorisieren, Rotation, Entrastern, Kantenverschärfung	je nach SW
Speichererw. erforderlich/ wieviel?	empfohlen 2 MByte RAM	ja/2 MByte RAM	nein
Systemvoraussetzungen	IBM u. Kompatible, Apple	AT u. Kompatible	IBM u. Kompatible mit XT-, AT-Bus
Bemerkungen	Anschluß an DEC-Ethernet in Vorbereitung	auch für Apple Macintosh für 12 175,– DM; in Vorb. Anschluß an DEC-Ethernet	

MARKTÜBERSICHT



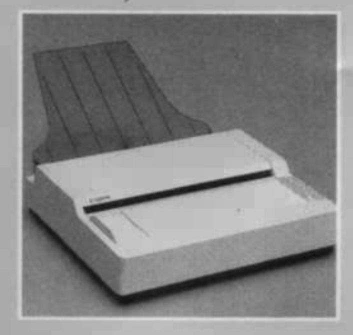


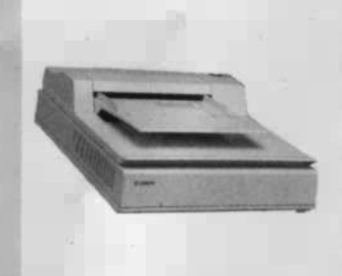




Berthold/Berthold	Cameron/Cameron	Cameron/Cameron	Cameron/Cameron
Berthold LOGO/Rollen	HS Typ 3/Amiga	Handy-Sc. Typ 3/IBM/Rollen	Handy-Sc. Typ 3/Atari/Rollen
8550,–	898,–	898,–	898,–
k. A.	SW, OCR-SW, Handbuch	SW, Interface, Handbuch	SW, Handbuch, Interface
CCD	CCD	CCD	CCD
seriell	Steckkarten-Module	Steckkarte	Steck-Modul
WS/Data Box	512 KByte/8 MByte	keine	512 KByte
k. A.	k. A.	k. A.	k. A.
ja/ja	ja/nein	ja/nein	ja/nein
Linien, Halbton	Linien, Dither	Linien, Dither	Linien, Dither
216 × 355 mm/76 × 76 mm	je nach RAM	k. A.	je nach Speicher
12–64 Linien/mm	200, 300, 400	200, 300, 400	200, 300, 400
ja/A6, A5, A4	ja/Scanlänge in cm	ja/64 mm breit, Länge wählb.	ja/64 mm × frei
17/ja	16/ja	16/ja	16/ja
nein/k. A.	als Grautöne	nein/keine	nein/keine
ja	ja	ja	ja
ja/mit Box	ja/Maus	ja/Maus	ja/Maus
ja/ja	ja/ja	ja/ja	ja/ja
ja/beliebig ja/beliebig	ja/beliebig ja/beliebig	ja/je nach SW ja/bis 25 %	ja/beliebig ja/beliebig
keine	Amiga-DOS	MS-DOS ab 2.0	Atari-Betriebs-Syst.
PIXEL	TIFF	PC Paint Brush, PCX, fast alle durch Treibertechn.	DEGAS
ja (nicht angeboten)	ja/5 + lernfähig	ja/5 + lernfähig	ja/5 + lernfähig
Pixel-Grafik, Logotype	alle üblichen	alle üblichen	alle üblichen
nein	empf. 1 MByte	nein	empfohlen 1 MByte
Workstation, Grafikstation	Amiga 500, 1000, 2000	IBM XT, AT + Komp., 512 KByte RAM	Atari 520 ST/ 1040 ST/Mega ST
	OCR- u. Grafik-SW werden standardmäßig mitgelief.	Lieferung inkl. Grafik- Paket, OCR-Softw., Treiber-Software	Grafik- u. OCR-Software gehören z. Lieferumfang

Scanner im Vergleich







Anbieter/ Hersteller	Canon/Canon	Canon/Canon	Computer 2000/Dest. Corp.
Name/ Scannertyp	IX-12/Durchzug (Rollen)	IX-12F/Flachbett	PC Scan 2010/2030/Rollen
Preis inkl. MwSt.	3979,–	4332,–	6368,–
Lieferumfang	Scanner, Sc.-Treiber	Scanner-Treiber	k. A.
Technologie/Sensor	CCD, Prozessor 37B05	CCD, Prozessor 37B05	CCD, Prozessor 80186
Schnittstellen	seriell	seriell	SCSI
Speicherkapazität min./max.	k. A./k. A.	k. A./k. A.	k. A./750 KByte
Abtastgeschwindigkeit 1 S. A4	12 s	16 s	10 s inkl. Transfer
Übertragung byteweise/kompr.	ja/je nach Software	ja/je nach Software	k. A./nein
Arbeitsmodi	Dither	Dither	Linien, Dither, Halbton
Vorlagengröße max./min.	A3 Hoch, Letter quer/k. A.	A4 Letter/Legal/k. A.	22 × 36 cm/5 × 8 cm
Auflösung in dpi (Stufen)	je nach Software	je nach Software	70–300
Scanfläche einstellbar/Formate	je nach Software	je nach Software	ja/A4, Legal
Grauwerte/Wechsel möglich	je nach Software	je nach Software	256/nein
Farbe/Blindfarben	je nach Software	je nach Software	nein/k. A.
Bildschirmanzeige der gescannten Vorlage	je nach Software	je nach Software	ja
Ausschnitte festlegen/wie?	je nach Software	je nach Software	ja/Maus, Fenster
Ausschnitte getrennt bearbeiten?/speichern	je nach Software	je nach Software	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	je nach Software je nach Software	je nach Software je nach Software	ja/bis 1600 % k. A.
Software-Voraussetzungen	keine	k. A.	Windows, Apple TOS
Grafikdateiformate	k. A.	je nach Software	TIFF, GEM, PC Paint, PC Paint Brush, EPSF
OCR-fähig/Zahl der Schriften	ja/ca. 12	ja/ca. 12	ja/15
Graf. Editierfunktionen	je nach Software	je nach Software	Schneiden, Drehen, Füllen usw.
Speichererw. erforderlich/ wieviel?	je nach Software	je nach Software	ja/k. A.
Systemvoraussetzungen	XT, AT und Kompatible	XT, AT und Kompatible	IBM PC, XT, AT, PS/2-30, 60, 80, Macintosh
Bemerkungen			



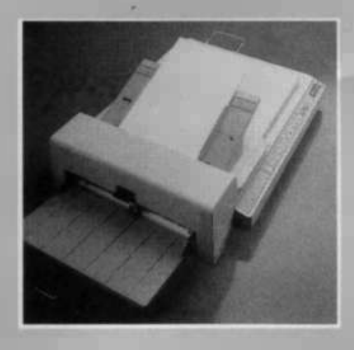






Computer 2000/Dest. Corp.	C-COM/C-COM	DFI/Daimond Flower	Epson/Epson
PC Scan 1010/1030/Flachbett	Scan Tron 300/Flachbett	HS-2000/Rollen	Scanner-Kit = Aufsatz
6648,–	3675,–	513,–	598,–
k. A.	SCSI-Interface	Software	inkl. Software
CCD, Prozessor 80186	CCD, Prozessor: NC2448	CCD, Proz.: Custom Chip	Silizium-Fotodiode
SCSI	SCSI	HS ist Schnittstelle	seriell
750 KByte	k. A.	384 KByte	keine
20 s inkl. Transfer	19 s, 22 s Transfer	2 cm/s	27 Zoll/s
k. A./ja	ja/ja	ja/nein	nein/nein
Linien, Dither, Halbton	Dither, Halbton	Linien, Dither, Halbton	Linien
22 × 36 cm, beliebig	Legal, k. A.	Scanfläche 105 mm	je nach Druckformat
70–300	75–300 in 7 Stufen	200	60–180
ja/A4, Legal	ja/variabel	ja	ja/wie Drucker
16/ja	16/ja	16/nein	nein
nein/k. A.	nein/wählbar	nein/rot	nein/keine
ja	ja	ja	ja
ja/Maus, Fenster	ja/Maus	ja/k. A.	ja/Software
ja/ja	ja/ja	ja/ja	ja/ja
ja/bis 1600 % k. A.	ja/4 × Zoom k. A.	ja/k. A. ja/k. A.	ja/k. A. ja/k. A.
Windows, Apple TOS	keine	HALO DPE dabei	Epscan, Layout inkl.
TIFF, PC, Paint, PC Paint Brush, EPSF	TIFF, GEM, PC Paint Brush	TIFF, GEM	Epscan, Layout, Windows
ja/15	ja/lernfähig	ja (Fremdanbieter)	nein
Zoom, Ausschneiden, Drehen usw.	Paintbrush-Schnittstelle	bis zu fatbit editing	alle üblichen
nein	nein, empf. 1 MByte	ja, falls hohe Aufl. bei Bearb.	nein
IBM PC, XT, AT, PS/2-30, 60, 80, Macintosh	AT ab 10 MHz	XT, AT + Komp., 512 KB RAM, MS-DOS ab 2.0, 1 DUA-Kanal	IBM PC + Komp., ab MS-DOS 2.11
		HS 2000 ist ein sog. „Handy-Scanner“	Das Kit ist ein Aufsatz für Epson-Drucker EX-800/100 u. LQ-2500/2550. Layout kann Text + Grafik mischen

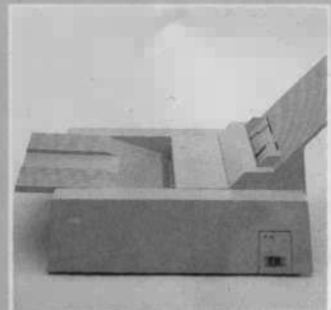
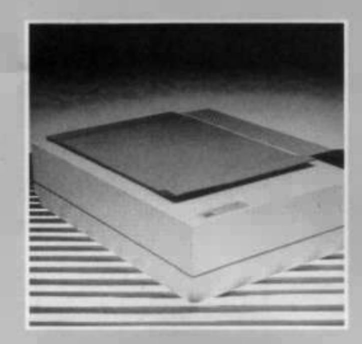
Scanner im Vergleich







Anbieter/ Hersteller	Fujitsu/Fujitsu	Hewlett Packard/HP	IBM Deutschland/IBM
Name/ Scannertyp	M3094 E/F/Flachbett	HP ScanJet/Flachbett	IBM 3117/Flachbett
Preis inkl. MwSt.	k. A.	5557,–	3055,–
Lieferumfang	k. A.	Schnittst., SW, Scanning Gallery	inkl. Handbuch
Technologie/Sensor	CCD	CCD	CCD
Schnittstellen	RS 232/binär (Videosignal)	parallel Centr.	PC-Adapter
Speicherkapazität min./max.	k. A.	speichert auf Festpl.	optional
Abtastgeschwindigkeit 1 S. A4	22 s	20 s inkl. Transfer	30 s inkl. Transfer
Übertragung byteweise/kompr.	ja/nein	k. A./k. A.	ja/ja
Arbeitsmodi	Linien/Halbton	Linien, Dither, Halbton	Linien
Vorlagengröße max./min.	B4/A5	216 × 356 mm/keine Begr.	216 × 297 mm
Auflösung in dpi (Stufen)	200/240/300/400	38–600	240 × 240
Scanfläche einstellbar/Formate	ja/A5–B4	ja/frei wählbar	ja/k. A.
Grauwerte/Wechsel möglich	64/k. A.	16/ja	17/ja
Farbe/Blindfarben	nein/k. A.	nein/gelb	ja/gelb, hellgrün
Bildschirmanzeige der gescannten Vorlage	ja (mit GDA-Software)	ja	ja
Ausschnitte festlegen/wie?	ja/Maus	ja/per Software	ja/k. A.
Ausschnitte getrennt bearbeiten?/speichern	ja (GEM)/ja	ja/ja	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/bis 100 % k. A.	ja/bis 800 % ja/bis 7 %	ja/theor. unbegrenzt ja/theor. unbegrenzt
Software-Voraussetzungen	optional GEM	Windows	DOS ab 2.1, Image Support Facility
Grafikdateiformate	TIFF, GEM, PC Paint, PC Brush, CITT 6.3 Datenkompression	TIFF, GEM, PC Paint, PC Paint Brush	TIFF, Image Data Stream, EPS
OCR-fähig/Zahl der Schriften	ja/49	ja/20	nein
Graf. Editierfunktionen	keine	k. A.	alle üblichen
Speichererw. erforderlich/ wieviel?	ja/mind. 2 MByte	nein	nein
Systemvoraussetzungen	AT und Kompatible	XT, AT, PS/2 mit 640 KB RAM	IBM PC + Komp., PS/2, IBM 3X, IBM AS400, IBM/370
Bemerkungen	Controller und Software von GDA/Bremen		


IBM Deutschland/IBM	IBM Deutschland/IBM	Kulkoni/AVR	Kyocera/Kyocera
IBM 3119/Flachbett	IBM 3118/Flachbett	AVR 302/Flachbett	A 400/Flachbett
3420,–	6555,–	5500,–	ca. 4000,–
k. A.	inkl. Handbuch	k. A.	Scan-Software, Interface
CCD	CCD	CCD	CCD
ser. + par. opt., PC-Adapter	PC-Adapter, opt. parallel	37pol. Video-Interface	AT-Interface-Karte
keine	keine	k. A.	k. A.
30 s inkl. Transfer	12 s inkl. Transfer	9 s inkl. Transfer	21 s
ja/ja	ja/ja	ja/k. A.	spezielle Übertr.
Linien, Dither, Halbton	Linien	Linien, Dither, Halbton	Linien, Halbton
216 × 297 mm	216 × 297 mm	8,5 × 11,7 inch/k. A.	A4/beliebig
300 × 300	240 × 240	300	75 × 75 bis 800 × 800 stufenlos
ja/k. A.	ja/k. A.	nein	ja/beliebig
128/ja	17/ja	32/nein	16/ja
ja/gelb, hellblau	ja/gelb, hellgrün	nein/k. A.	nein/31 Halbtöne
ja	ja	ja	ja
ja/k. A.	ja/k. A.	ja/Ausschnitt-Wahl	ja/Grafik, Maus
ja/ja	ja/ja	ja/ja	ja/ja
ja/theor. unbegrenzt ja/theor. unbegrenzt	ja/theor. unbegrenzt ja/theor. unbegrenzt	ja/bis A4 stufenlos k. A.	ja/k. A. ja/k. A.
DOS ab 2.1, Image Support Facility	ab DOS 2.1, Image Support Facility	eigene Software	Windows oder GEM, Ventura usw.
TIFF, Image Data Stream, EPS	TIFF, Image Data Stream, EPS	TIFF, GEM, PC Paint, ASCII, FAX, BIN, PIC, MSI	TIFF
nein	nein	ja/4, lernfähig	nein (geplant)
alle üblichen	alle üblichen	je nach Software	k. A.
nein	nein	nein	nein
PS/2, IBM/3X, IBM/AS 400, IBM/370	IBM PC, PS/2, IBM/3X, IBM/AS400, IBM/370	IBM AT u. Komp., IBM PS/2	MS-DOS
		1,3 MByte auf Mega-Buffer, schnelles Zoomen + Bearbeiten	lieferbar vorauss. ab 4. Quartal 88

Scanner im Vergleich



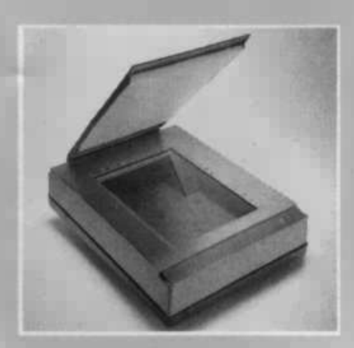


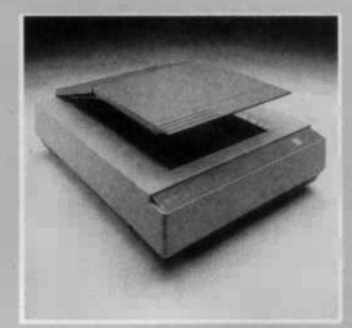


Anbieter/ Hersteller	Lasersoft/Pixelogic	Lasersoft/Pixelogic	Lasersoft/Truvel
Name/ Scannertyp	ProViz S-W/Video	ProViz Color/Video	TZ-3/Flachbett
Preis inkl. MwSt.	4389,–	7410,–	20 520,–
Lieferumfang	Contrl., Interf., Kabel, Software	Contr., Interf., Kabel, Software	SCSI oder GPIC
Technologie/Sensor	A/D-Wandlung	Video-Scanner	CCD
Schnittstellen	SCSI	SCSI	SCSI, parallel
Speicherkapazität min./max.	150 KByte/300 KByte	150 KByte/930 KByte	k. A.
Abtastgeschwindigkeit 1 S. A4	Echtzeit	5 s	10 s inkl. Transfer
Übertragung byteweise/kompr.	–/ja	–/ja	ja/ja
Arbeitsmodi	Linien, Dither, Halbton	Linie, Dither, Halbton	Linien, Dither, Halbton
Vorlagengröße max./min.	jede, da durch Optik best.	best. durch Optik	A3/4 × 4 cm
Auflösung in dpi (Stufen)	bis 600	75–6000	300–900
Scanfläche einstellbar/Formate	bestimmt durch Optik	ja/je nach Optik	ja/4 × 4–A3
Grauwerte/Wechsel möglich	256/ja	256/ja	256/ja
Farbe/Blindfarben	nein/k. A.	ja/keine	nein/k. A.
Bildschirmanzeige der gescannten Vorlage	ja	ja	ja
Ausschnitte festlegen/wie?	ja/Optik	ja/Optik	ja/Software
Ausschnitte getrennt bearbeiten?/speichern	ja/ja	ja/ja	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/unbegrenzt ja/unbegrenzt	ja/unbegrenzt ja/unbegrenzt	ja/unbegrenzt ja/unbegrenzt
Software-Voraussetzungen	Mac-Betriebs-Syst.	Mac-Betr.-System	Windows, Mac-BS
Grafikdateiformate	TIFF, RIFF, EPSE, MacPaint, ProViz	TIFF, RIFF, EPSH, Mac Paint, PICT, ProViz	TIFF, PC Paint, PICT
OCR-fähig/Zahl der Schriften	k. A.	nein	nein
Graf. Editierfunktionen	k. A.	keine	keine
Speichererw. erforderlich/ wieviel?	ja/mind. 2 MByte	ja/bis 8 MByte	ja/8 MByte
Systemvoraussetzungen	Apple-Mac Plus, Mac SE, Mac II	Apple Mac Plus, Mac SE, Mac II	IBM AT, Macintosh II
Bemerkungen			



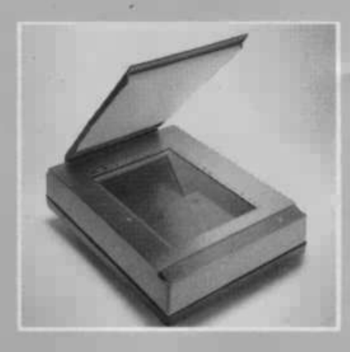




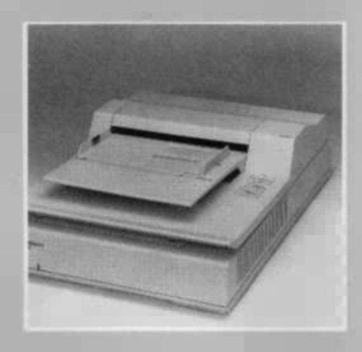


Lasersoft/Truvel	Macrotron/Datacopy	Macrotron/Datacopy	Macrotron/Datacopy
TZ-3WC/Flachbett	Modell 830/Flachbett	Jet Reader/Einzug	Modell 730/Flachbett
33 060,–	6122,–	2274,–	4514,–
Interf., Software	k. A.	k. A.	k. A.
CCD	CCD	CCD	CCD
SCSI, parallel	SCSI, parallel (eigene)	SCSI, parallel (eigene)	SCSI, eigene
k. A.	keine	keine/keine	keine
80 s	12 s inkl. Transfer	12 s inkl. Transfer	12 s inkl. Transfer
k. A.	ja/ja (je nach Software)	ja/ja	ja/ja
Linien, Halbton	Linien, Dither, Halbton	Linien, Dither, Halbton	Linien, Dither, Halbton
A3/4 × 4 cm	k. A./0 × 0 cm	A4/A6	k. A./0 × 0 cm
300–600/24 Bit	75 × 75–300 × 300	150, 200, 240, 300	150, 200, 240, 300
ja/A3–4 × 4 cm	ja/22 × 31 cm	ja/22 × 31 cm	ja/22 × 31 cm
256/k. A.	64/ja	16/ja	16/k. A.
16 Mio. Farben/keine	nein/k. A.	nein/k. A.	nein/k. A.
ja	je nach Software	ja	ja
ja/k. A.	je nach Software	ja/je nach Software	ja/je nach Software
ja/ja	je nach Software	ja/ja	ja/ja
ja/unbegrenzt ja/unbegrenzt	ja/je nach Software ja/je nach Software	ja/je nach Software ja/je nach Software	ja/je nach Software ja/je nach Software
Windows, Mac-BS	je nach Software	je nach Software	keine
TIFF, PC Paint, PICT	je nach Software	je nach Software	je nach Software
nein	ja/20–30 (lernfähig)	ja/20–30 (lernfähig)	ja/20–30 (lernfähig)
keine	keine	keine	keine
ja/8 MByte	je nach Software	je nach Software	je nach Software
IBM XT, AT, Macintosh, Apollo, Sun	XT, 512 KB RAM, 10 MB Festpl.	PC, XT, 512 KB RAM, 10 MB Festpl.	XT, 512 KB RAM, 10 MB Festpl.
	auch für Apple verfügbar	auch für Apple verfügbar	auch für Apple verfügbar

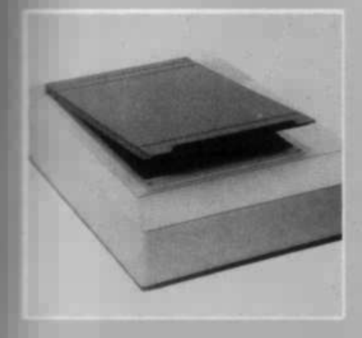
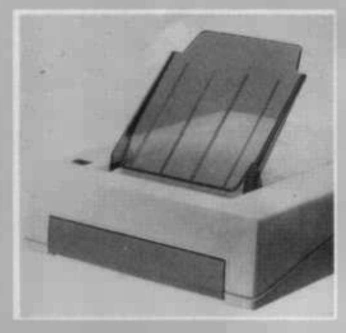
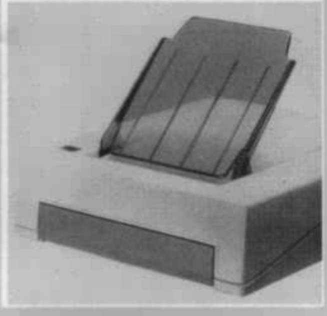
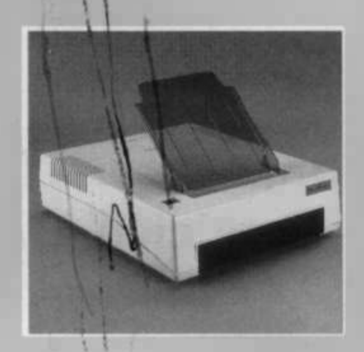
Scanner im Vergleich







Anbieter/ Hersteller	Macrotron/Datacopy	Macrotron/Princeton	Mannesmann-Kienzle/MK
Name/ Scannertyp	Modell 840/Flachbett	LS 300/Rollen	Scanner 161/Flachbett
Preis inkl. MwSt.	15 931,–	3363,–	6726,–
Lieferumfang	k. A.	k. A.	Interface
Technologie/Sensor	CCD	CCD	CCD
Schnittstellen	SCSI, parallel (eigene)	parallel (eigene)	parallel
Speicherkapazität min./max.	keine	keine	2 × 16 KByte
Abtastgeschwindigkeit 1 S. A4	ca. 4 min. inkl. Transfer	15 s inkl. Transfer	20,4 s inkl. Transfer
Übertragung byteweise/kompr.	ja/ja	ja/ja	ja/nein
Arbeitsmodi	Halbton	Linien, Dither	Dither/Halbton
Vorlagengröße max./min.	k. A./0 × 0 cm	A4/A6	216 × 297/140 × 140
Auflösung in dpi (Stufen)	400	75–300	38–600
Scanfläche einstellbar/Formate	ja/22 × 31 cm	ja/diverse	ja/beliebig
Grauwerte/Wechsel möglich	256/nein	16/ja	16/ja
Farbe/Blindfarben	nein/k. A.	nein/k. A.	nein/k. A.
Bildschirmanzeige der gescannten Vorlage	je nach Software	je nach Software	ja
Ausschnitte festlegen/wie?	je nach Software	je nach Software	ja/Software
Ausschnitte getrennt bearbeiten?/speichern	je nach Software	je nach Software	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/je nach Software ja/je nach Software	je nach Software je nach Software	ja/bis Faktor 2 k. A.
Software-Voraussetzungen	keine	keine	MK-SW: Kiscan eigene
Grafikdateiformate	je nach Software	je nach Software	ja/bis zu 20 die üblichen
OCR-fähig/Zahl der Schriften	nein	ja/lernfähig	
Graf. Editierfunktionen	keine	je nach Software	je nach Software
Speichererw. erforderlich/ wieviel?	je nach Software	nein	MS-DOS-Rechner,
Systemvoraussetzungen	XT, 512 KB RAM, 10 MB Festpl.	XT, 512 KB RAM, 10 MB Festpl.	MK-Rechner WS 9008
Bemerkungen	auch für Apple verfügbar		

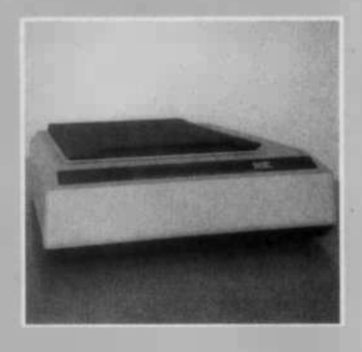





Mannesmann Tally/MT	Mannesmann Tally/MT	Mannesmann Tally/MT	Microplex/Microtex
MT Scanner FC/Flachbett	MT Scanner A/Rollen	MT Scanner C/Rollen	DocuMaster 1100/Rollen
5680,–	5680,–	4548,–	9800,–
k. A.	k. A.	k. A.	DocuRead + DocuScan Software
CCD	CCD	CCD	fotoelektr.
seriell	parallel, seriell	seriell	parallel, seriell
k. A.	k. A.	k. A.	keine
10 s	10 s	10 s	14 s inkl. Transfer
k. A.	k. A.	k. A./k. A.	ja/nein
Linien, Halbton	Linien, Halbton	Linien, Halbton	Linien, Dither, Halbton
8,5 × 14 Zoll/3 × 3 Zoll	8,5 × 14 Zoll/3 × 3 Zoll	8,5 × 14 Zoll/3 × 3 Zoll	A4/k. A.
75–300 in 7 Stufen	75–300 in 7 Stufen	75–300 in 7 Stufen	300 × 300
ja/A4–216 × 356 mm	k. A.	k. A.	ja durch Rahmen/k. A.
16/k. A.	64/k. A.	16/nein	ja/ja
nein/k. A.	nein/k. A.	nein/k. A.	nein/k. A.
ja	ja	ja	ja
k. A.	k. A.	k. A.	ja/Rahmen
k. A.	k. A.	k. A.	nein/nein
ja/k. A. ja/bis 1 %	nein ja/bis 1 %	k. A. k. A.	ja/k. A. ja/k. A.
k. A.	k. A.	k. A.	keine
TIFF	TIFF	TIFF	GEM, PC Paint, PC Paint Brush, je nach Programm
nein	nein	nein	ja/über 100
k. A.	k. A.	k. A.	je nach Programm
nein	nein	nein	ja/1 MByte (außer bei OCR)
IBM PC, Apple Macintosh	IBM PC, Apple Macintosh	IBM PC, Apple Macintosh	XT, AT und Kompatible

Scanner im Vergleich

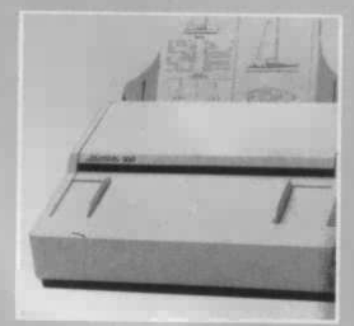






Anbieter/ Hersteller	Microplex/Microtex	Nixdorf/Ricoh	Olivetti/Olivetti
Name/ Scannertyp	DocuMaster 1200/Flachbett	EL 08/Rollen	PS 103/Flachbett
Preis inkl. MwSt.	9800,–	5278,–	9114,–
Lieferumfang	Interface, Software	Karte + Software	Karte, SW: Eyestar, Greyscan
Technologie/Sensor	fotoelektr.	k. A.	k. A.
Schnittstellen	seriell üb. DMA-Interface	parallel (bidirektional)	parallel
Speicherkapazität min./max.	keine	k. A./2 MByte	k. A.
Abtastgeschwindigkeit 1 S. A4	14 s inkl. Transfer	9 s	11 s
Übertragung byteweise/kompr.	ja/nein	k. A./k. A.	k. A./ja
Arbeitsmodi	Linien, Dither, Halbton	Linien, Dither, Halbton	Linien, Halbton
Vorlagengröße max./min.	A4/k. A.	364 × 257 mm/105 × 148 mm	8,5 × 14 Zoll/k. A.
Auflösung in dpi (Stufen)	300 × 300	180, 200, 240, 300	max. 300
Scanfläche einstellbar/Formate	ja durch Rahmen	ja/frei wählbar	ja/stufenlos
Grauwerte/Wechsel möglich	10/ja	16/k. A.	256/k. A.
Farbe/Blindfarben	nein/k. A.	nein/gelb	nein/k. A.
Bildschirmanzeige der gescannten Vorlage	ja	ja	ja
Ausschnitte festlegen/wie?	ja/Rahmen	ja/Markieren	ja/Maus, Cursor
Ausschnitte getrennt bearbeiten?/speichern	nein/nein	nein/ja	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/k. A. ja/k. A.	ja/bis 32 : 1 ja/bis 1 : 32	ja/k. A. ja/k. A.
Software-Voraussetzungen	keine	Windows	Windows
Grafikdateiformate	GEM, PC Paint, PC Paint Brush, je nach Software	TIFF, PC Paint Brush, Windows Paint	TIFF
OCR-fähig/Zahl der Schriften	ja/über 100	nein	optional
Graf. Editierfunktionen	programmabhängig	Airbrush und alle üblichen	Cut, Copy, usw.
Speichererw. erforderlich/ wieviel?	ja/1 MByte (nicht bei OCR)	nein	nein
Systemvoraussetzungen	XT, AT u. Kompatible	k. A.	80286- u. 80386-Rechner
Bemerkungen			

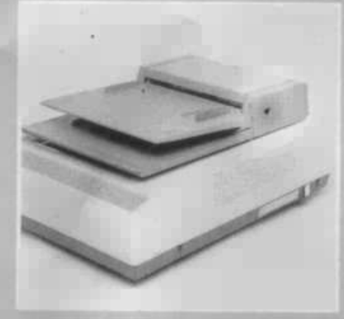


Olivetti/Olivetti	Panasonic/Panasonic	Philips/Mikrotec	Print Partner/QMS
PS 102/Flachbett	EX-RS 505/Flachbett	MS 300A/Rollen	IS 300/Rollen
5124,–	3490,–	4995,–	4990,–
Karte, Software Eyestar	Software, Interface-Karte, Kabel	Interface-Karte, Software, Kabel	inkl. PC-Interface
k. A.	CCD	CCD	CCD
parallel	Interface-Karte	parallel	PC-Interface
k. A.	k. A.	k. A.	keine
9,9 s	3 m/s	10 s/k. A.	12 s inkl. Transfer
k. A./ja	ja/nein	ja/nein	high speed Video-Interface
Linien, Halbton	Linien, Dither	Linien, Dither, Halbton	Dither, Halbton
8,5 × 14 Zoll	B4/A5 und kleiner	8,5 × 14 inch/3 × 3 inch	210 × 297 mm
max. 300	200, 300, 400	75–300	max. 300, je nach Software
ja/stufenlos	ja/A5, A4, B5, B4, Letter	ja/stufenlos	ja/k. A.
64/k. A.	16/ja	16/ja	32/ja
nein/7/k. A.	nein/hellgrün	nein/cyan, gelb	nein/k. A.
ja	ja	ja	ja
ja/Maus, Cursor	ja/Cursor, Maus	ja/Software	ja/k. A.
ja/ja	ja/ja	ja/ja	ja/ja
ja/k. A.	ja/bis Pixelebene	ja/k. A.	ja/k. A.
ja/k. A.	ja/k. A.	ja/k. A.	ja/k. A.
Windows	keine	Windows, Eyestar	MS-DOS
TIFF, PC Paint, PC Paint Brush	TIFF, PCX, ASCII, PC Paint, Paint Brush	TIFF, PC Paint, PC Paint Brush, Eyestar	PC Paint Brush
optional	ja/lernfähig	nein	ja opt./12
wie üblich	Vergr., Verkl., Radieren, Ausschn. bearb., Schattieren	Malen, Kontraste einsetzen, wegnehmen, Linien usw.	keine
nein	nein	nein	nein
8086-, 80286- u. 80386-PC	IBM PC und Komp., Atari	AT und Kompatible	IBM PC, XT, AT und Kompatible

Scanner im Vergleich







Anbieter/ Hersteller	Print Partner/QMS	Ricoh/Ricoh	Sharp/Sharp
Name/ Scannertyp	IS 300F/Flachbett	RS 311/Flachbett	JX-450/Flachbett
Preis inkl. MwSt.	7990,–	5529,–	19 950,–
Lieferumfang	PC-Interface	Interface ISI 8, Software	Farb-Scan-Software
Technologie/Sensor	CCD	CCD, Prozessor 8 Bit	CCD
Schnittstellen	PC-Interface	parallel, bidirektional	SCSI, GPIB/IEEE-488
Speicherkapazität min./max.	keine	keine	je nach EDV-System
Abtastgeschwindigkeit 1 S. A4	12 s inkl. Transfer	14 s/k. A.	102 s bei 300 dpi
Übertragung byteweise/kompr.	high speed Video-Interface	nein/nein	ja/ja
Arbeitsmodi	Dither, Halbton	Linien, Dither, Halbton	Halbton
Vorlagengröße max./min.	210 × 297 mm/50 × 50 mm	216 × 297/keine Begrenz.	296 × 435 mm/3 × 3 mm
Auflösung in dpi (Stufen)	max. 300, je nach Software	180, 200, 240, 300	30–300 in Einerschritten
Scanfläche einstellbar/Formate	ja	ja/beliebig	ja/beliebig
Grauwerte/Wechsel möglich	32/ja	16/ja	ja/k. A.
Farbe/Blindfarben	nein/k. A.	nein/gelb, hellblau	ja/k. A.
Bildschirmanzeige der gescannten Vorlage	ja	ja	ja
Ausschnitte festlegen/wie?	ja/k. A.	ja/Fensterdefinition	ja/Maustreiber
Ausschnitte getrennt bearbeiten?/speichern	ja/ja	ja/ja	ja/ja
Lassen sich eingescannte Vorlagen am Bildschirm vergrößern/max. verkleinern/min.	ja/k. A. ja/k. A.	ja/beliebig ja/beliebig	nein nein
Software-Voraussetzungen	MS-DOS	Windows	k. A.
Grafikdateiformate	PC Paint Brush	TIFF, PC Paint Brush, MS-Paint	TIFF, GEM, PCX, TAG
OCR-fähig/Zahl der Schriften	ja opt./12	ja/unbegrenzt	nein
Graf. Editierfunktionen	keine	alle üblichen	k. A.
Speichererw. erforderlich/ wieviel?	nein	nein	nein
Systemvoraussetzungen	IBM PC, XT, AT und Kompatible	IBM XT, AT u. Kompatible, PS/2 in Vorb.	Großen Speicher: 1 A3-Farbbild benötigt 50 MByte in 300 dpi
Bemerkungen			

MARKTÜBERSICHT

			
Siemens/Siemens	Silver Reed/Silver Reed	Silver Reed/Silver Reed	Softline/Softline
ST 400/Flachbett	SP 10/Flachbett	SPAT/Rollen	ShapeScan/k. A.
10 214,–	2495,–	1995,–	2598,–
Scan-Software	Software, Handbuch	Software, Handbuch	Scanner, Kabel, Software
CCD, Prozessor: 80188	CCD, Spiegel/Optik	Spiegel/Optik, CCD	CCD
SCSI	parallel	parallel	parallel
2 MByte	640 KByte/k. A.	1 MByte/k. A.	640 KByte/2 MByte
9 s b. 400 dpi, <5 s Transfer	10 s, Transfer 10 s	10 s/Transfer 10 s	38 s
ja/nein	ja/k. A.	ja/k. A.	k. A./k. A.
Linien, Dither, Halbton	k. A.	k. A.	Linien, Dither
225 × 310 mm/beliebig	210 × 297 mm	210 × 297 mm/k. A.	A4/unbegrenzt
3, 200, 400	200	200	75–300
ja/A4, A5, A6, Letter	nein	nein	ja/wählbar
64/ja	16/ja	16/ja	16/ja
nein/gelbgrün	nein/k. A.	nein/k. A.	nein/k. A.
ja	ja	ja	ja
ja/Maus	ja/Bildfenster	ja/Bildfenster	ja/über Software
ja/ja	ja/ja	ja/ja	ja/ja
ja/bis 3200 % ja/bis 25 %	ja/100 % ja/100 %	ja/100 % ja/100 %	ja/bis A4 k. A.
Windows, Apple Mac, Unix	keine	k. A.	keine
TIFF, GEM, RIFF, EPSF, PSF, PICT	PC Paint Brush, IMG	Image IMG, DEGAS PI 1/Pi, DIN, BIT Forma, Option	TIFF, GEM, PC Paint, PC Brush ASCII, Postscript, CUT
nein	nein	ja/k. A.	ja/lernfähig
k. A.	Spiegel, Invertieren	Radieren, Spiegeln, Ausschneiden, Lupe, Kreise, Linie, Box, Füllen	k. A.
ja/2 MB im PC	nein	nein	nein
AT u. Kompatible, 386, Mac SE, MAC II, Apollo, WS 30	IBM PC und Kompatible	Atari ST Serie	PC ab 640 KByte RAM
			Grafikkarte (CGA, EGA, VGA, Hercules o. a.)

Wilfried Voss

Der Stream-Editor

Ein neues DOS-Filter

Als 'Filter' wird in der Datenverarbeitung ein Programm bezeichnet, das Daten von einer Standardeingabe liest, die Daten ändert und das Ergebnis auf die Standardausgabe schreibt: Die Daten sind 'gefiltert' worden. Unter MS-DOS kennt man standardmäßig nur die Filter SORT, FIND und MORE [1].

Unter dem Betriebssystem Unix ist der Stream-Editor 'sed' bekannt, mit dem es möglich ist, Manipulationen an Zeichenketten mit einer Anweisungsliste (Script) durchzuführen. Um dieses nützliche Software-Werkzeug auch unter MS-DOS verfügbar zu haben, wurde das folgende, in C geschriebene Programm entwickelt, das sich allerdings nur zum Teil an die von UNIX geforderte Syntax hält.
Für die Programmentwicklung stand der Microsoft-C-Compiler in den Versionen 3.00 und 4.00 auf einem IBM-kompatiblen XT zur Verfügung. Viele Anregungen zu dem folgenden Programm wurden aus [2] übernommen, in dem ein Textverarbeitungssystem in C beschrieben ist.

Was kann der Stream-Editor?

Der Stream-Editor sucht, ersetzt oder löscht Zeichenketten (Strings) in einer anzugebenden ASCII-Datei, bewegt einen fiktiven Cursor und druckt auf Wunsch einzelne Zeilen oder die gesamte Datei ab 'Cursorposition' – wahlweise mit Zeilennumerierung – aus.
Die Anweisungen erhält das Programm über die Standard-Eingabe (Grundeinstellung: Tastatur), oder, wenn diese mit „<“ in der Kommandozeile entsprechend umgeleitet wurde, aus einer Datei (Script), die eine Liste mit Anweisungen enthält. Das Script ist also eine Folge von Editierkommandos.
Der Vorteil gegenüber einem Editor liegt in der wesentlich größeren Geschwindigkeit, mit der die Anweisungen, zudem programmgesteuert, durchgeführt werden. Ähnliche Zeichenmanipulationen führen z. B. der Preprozessor der Programmiersprache C (define-Anweisung) sowie alle Makro-Assembler durch.
In *Tabelle 1* sind alle möglichen Befehle und deren Optionen aufgeführt. Im Programm ist eine kleine Hilfe dazu vorgesehen: Mit Eingabe von 'sed ?' erscheint diese Liste mit Hinweisen zur Syntax (*Bild 1*). Das gefilterte Ergebnis kann auf Wunsch ausgegeben werden, wobei wiederum die Möglichkeit besteht, die Standardausgabe

```
Dateiname :       sed.hlp

Syntax    :       sed source <stdin >stdout

c /+              Cursor rechts
c /-              Cursor links
c /^              Cursor hoch
c /v              Cursor runter
c /<              Cursor auf Zeilenanfang
c />              Cursor hinter Zeilenende
c /h              Cursor auf Textanfang
c /e              Cursor auf Textende

d                 Aktuelle Zeile löschen

f /string/pgd+    string suchen
                  p = Zeile ausgeben
                  g = Suchen bis File-Ende
                  d = Zeile löschen
                  + = Cursorposition um 1 erhöhen

k                 Zeichen in Cursorposition löschen

p /gn             Zeile ausgeben
                  g = Ausgabe bis File-Ende
                  n = Zeilennummer ausgeben

s /s1/s2/pg       Ersetze s1 durch s2
                  p = Zeile ausgeben
                  g = Ersetzen bis File-Ende

{                 Makro-Anfang
}g                Makro-Ende
                  g = Ausführen bis File-Ende
}99               Makro-Ende
                  Makro 99 mal ausführen
```

Bild 1. Der Hilfstext findet sich in der Datei SED.HLP

(Grundeinstellung: Bildschirm) umzuleiten, z. B. auf die Festplatte. Die Kommandozeile sieht demnach folgendermaßen aus:

sed textdatei <script >ausgabe

Die Angaben 'script' und 'ausgabe' sind optional. Ohne Angaben werden, wie schon erwähnt, Tastatur und Bildschirm verwendet.

Die Funktion

An einigen Beispielen läßt sich die Funktionsweise des Stream-Editors am einfachsten darstellen. Als zu filternde Datei wird der Quelltext des Stream-Editors „sed.c“ (*Bild 2*) gewählt. Ein erstes Script dazu zeigt *Tabelle 2:* Nach dem Aufruf „sed sed.c“ in der Kommandozeile kann das Script in dieser Form per Tastatur eingegeben werden, nachdem der Quelltext von der Festplatte oder Diskette geladen wurde. Als Abschluß des Scripts bei Tastatureingabe gilt eine Leerzeile mit abschließendem <Return>.
Einfacher ist es allerdings, diese Befehlsfolgen in einer Textdatei (z. B. mit dem Namen „script“) abzulegen, damit Falscheingaben korrigiert werden können. Manchmal kommt es auch vor, daß man verschiedene Text-Dateien nach der gleichen Vorschrift manipulieren möchte, ohne die ganze Liste noch einmal eingeben zu müssen. Die Kommandozeile lautet also nun:

sed sed.c <script

Das Ergebnis wird entsprechend *Tabelle 3* auf dem Bildschirm dargestellt.
Wie erwähnt, kann auch die Ausgabe des Ergebnisses umgeleitet werden. Zum Beispiel:

sed sed.c <script >newfile
Ablegen in einer Datei,

sed sed.c <script >prn
Ausgabe auf Drucker,

sed sed.c <script >com1
Serielle Ausgabe.

Einen Nachteil enthält das Script: Sollte die Zeichenkette „getparam(“ nicht im Quelltext vorkommen, werden die nachfolgenden Anweisungen trotzdem ausgeführt. Der fiktive Cursor bleibt bei erfolglosem Suchen/Ersetzen an der bisherigen Position. Eine Ausgabe erscheint in diesem Fall nicht sinnvoll.
Abhilfe schafft hier die Definition eines 'Makros'. Ein Makro wird sofort abgebrochen, wenn eine Such- oder Ersetzoperation erfolglos war oder der fiktive Cursor

```c
/****************************************************************/
/* Dateiname : sed.c                                            */
/* Funktion  : Stream Editor fuer MS-DOS                        */
/****************************************************************/
/* Include - Files */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <memory.h>
#include <malloc.h>
#include "defines.h"    /* Definition der Konstanten           */

/* Globale Deklarationen                                        */
/* Struktur einer Zeile im Textspeicher                         */
typedef struct line
{
  struct line *vorgaenger;         /* Pointer auf Vorgaenger */
  struct line *nachfolger;         /* Pointer auf Nachfolger */
  char zeileninhalt[ZEILENLAENGE+1]; /* Zeileninhalt incl.\0 */
} LINE;

/* Definition und Tabelle fuer Kommandoverarbeitung             */
struct anweisung
{
        char character;
        int  (*funktion) ();
};

int  overflow,cursorposition,makro_flag;
char file_name[80];
char *par_1,*par_2,*par_3;
LINE *text_anfang,*text_ende,*aktuelle_zeile;
LINE *script_anfang,*script_zeiger,*script_ende;
FILE *file_pointer;

#include "befehle.c"  /* Struktur/Auflistung Scriptbefehle   */

/* Deklaration der benoetigten Funktionen                       */
int  tausche(),fread(),rdscript();
int  filter(),suchen(),getparam(),rdhelp();
char *cut();
LINE *rdfile();

/**************************** main ****************************/
/* Aufruf: sed filename <script >stdout                        */
/**************************************************************/
main( argc, argv )

int  argc;
char *argv[];
{
    int rc;          /* returncode                  */

    /* Uebergebene Argumente auswerten */
    if (argc == 2 && strcmp(argv[1],"?") == NULL) rdhelp();
    else if ( argc == 2 )
        {
          strcpy(file_name,argv[1]);

          rc = fread();                          /* Sourcefile einlesen  */
          if ( rc == OKAY ) rc = rdscript();     /* Script einlesen      */
          script_ende = text_ende;               /* Pointer richten      */
          text_ende   = script_anfang->vorgaenger;
          if ( rc == OKAY ) filter();            /* Text veraendern      */
        }
        else printf("\nUngueltige Anzahl Parameter !\n");
}  /* end main */
/**************************************************************/
/* Funktionen zur Ausfuehrung der Scriptbefehle                */
/**************************************************************/
/* Funktion  : String suchen und tauschen */
substitute()
{
char *suchwort,*tauschwort,*flags;

  while( TRUE )
  {
```

Bild 2. Das Hauptprogramm SED.C

```c
     /* Parameter aus Scriptzeile entnehmen */
     getparam( script_zeiger->zeileninhalt );
     suchwort = par_1; tauschwort = par_2; flags = par_3;
     /* Flags von Blanks und Kommentar befreien */
     flags = cut( flags );
     if (*suchwort == 0) break;

     while( TRUE )
     {
       if (suchen(suchwort) == FEHLER) break;
       tausche( suchwort, tauschwort );
       if (strchr(flags,PRINT) != NULL)
          {
          printf("%s\n",aktuelle_zeile->zeileninhalt);
          }
       if (strchr(flags,GLOBAL) == NULL ) break;
     }
   if (strchr(flags,CURSOR_R) != NULL) cursorposition++;
   break;
   }
} /* end substitute */

/* Funktion  : String suchen */
finden()
{
int  found;
char *suchwort, *flags;

  while( TRUE )
  {
     /* Suchwort aus Scriptzeile entnehmen */
     getparam( script_zeiger->zeileninhalt );
     suchwort = par_1; flags = par_2;
     /* Flags von Blanks und Kommentar befreien */
     flags = cut( flags );
     if (*suchwort == 0) break;

     while( TRUE )
     {
       if ((found = suchen(suchwort)) == FEHLER) break;
       if (strchr(flags,PRINT) != NULL)
          {
          printf("%s\n",aktuelle_zeile->zeileninhalt);
          }
       if (strchr(flags,DELETE) != NULL) delete();
       if (strchr(flags,GLOBAL) == NULL)
         break;
       else
         if (strchr(flags,DELETE) == NULL) cursorposition++;
     }
  if (strchr(flags,GLOBAL) != NULL
                && found == OKAY
                && cursorposition > 0) cursorposition--;
  if (strchr(flags,CURSOR_R) != NULL
                   && found == OKAY
                   && cursorposition < ZEILENLAENGE-1) cursorposition++;
  break;
  }
  return;
  } /* end finden */

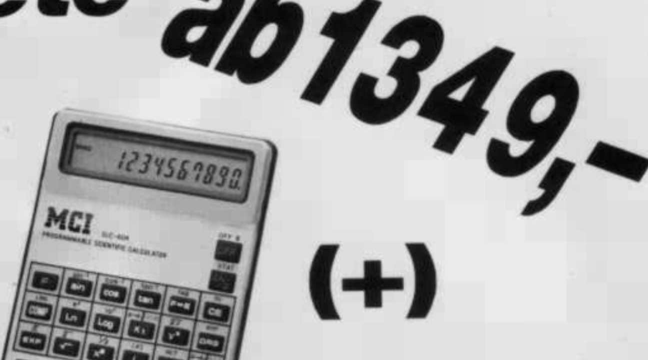
  /* Funktion  : Zeile loeschen */
  delete()
  {
  LINE *vo,*next;
  while( TRUE )
     {
       if (aktuelle_zeile->nachfolger == script_anfang) break;
       if (aktuelle_zeile == text_anfang)
        {
          text_anfang = text_anfang->nachfolger;
          aktuelle_zeile = text_anfang;
          break;
        }
       if (aktuelle_zeile == text_ende)
        {
          text_ende   = text_ende->vorgaenger;
          aktuelle_zeile = text_ende;
          break;
        }
```

KOMPLETTPREISE...SYSTEMPAKETE...1 JAHR GARANTIE...

KOMPLETTPREISE...SYSTEMPAKETE

System Pakete für kluge Rechner

ab 1349,-

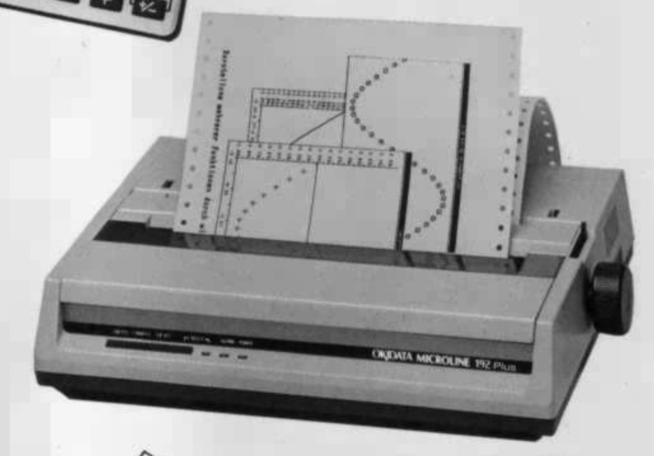




+

(+)

MCI Printer Plus **a. A.**
OKI ML192 Elite **a. A.**



System Paket 11

10 MHz

1349,-

- **MCI XT16SLC,** 640 K, 1 x 360 K, Clock, ser. par., 12" Monitor, Tastatur
- **MCI** Programmierbarer Taschenrechner

optional MCI 120 Zeichen Printer oder OKI ML 192 Elite

System Paket 21

10 MHz

1899,-

- **MCI XT16SLC,** 640 K, 1 x 360 K, Clock, ser. par., 12" Monitor, Tastatur
- **MCI** 20MB Festplatte
- **MCI** Programmierbarer Taschenrechner

optional MCI 120 Zeichen Printer oder OKI ML 192 Elite

System Paket 31



- **MCI AT4SLC,** 640 K, 1 x 1, 2 MB, Clock, ser. par., 12" Monitor, Tastatur
- **MCI** Programmierbarer Taschenrechner

optional MCI 120 Zeichen Printer oder OKI ML 192 Elite

System Paket 41



3199,-

- **MCI AT4SLC,** 640 K, 1 x 1, 2 MB, Clock ser. par., 12" Monitor, Tastatur
- **MCI** 20MB Festplatte
- **MCI** Programmierbarer Taschenrechner

opt. MCI 120 Zeichen Printer oder OKI ML 192 Elite

PREIS ... QUALITÄT ... 1 JAHR GARANTIE

KOMPATIBEL ... 24-STUNDEN-TEST ... LEISTUNG ... PREIS

MCI XT 16 SLC

alles drin!

Grundausstattung ohne Monitor

ab **899,–**

beinhaltet:

- voll IBM®XT kompatibel
- 8088 CPU + 8087 Sockel
- 8 XT Slots
- 256 KB freier Speicher
- 1 x 360 KB Floppy-Drive
- Color- oder Monochr. Grafikkarte (Hercules II komp. 720 x 348 P.)
- Deutsche Normtastatur MK 5111
- 150 W Schaltnetzteil
- Parallele Drucker-Schnittstelle



Erweiterungen für XT 16 SLC-Serie

2. Laufwerk 360 KB	249,–
Speichererweiterung auf 640 KByte	a. A.
Clock/Seriell-Karte	79,–
I/0 Plus II Karte	149,–
20 MB Festplatte mit XT-Controller	+ 589,–
30 MB Festplatte m. RLL XT-Contr.	+ 679,–
EGA-Set statt monochr. Karte	+ 1149,–
Opt. Maus mit Tablet	119,–
MS-DOS 3.3 deutsch + GW-Basic	+ 199,–
Professional Multifunktions-Tastatur MK 6000	+ 100,–
9" TTL Monitor grün	+ 150,–
12" Monitor grün od. bern.	+ 229,–
14" TTL Monitor grün, bern. od. weiß	+ 279,–





Grundausstattung ohne Monitor

ab **1899,–**

beinhaltet:

- voll IBM® AT kompatibel
- 80286 CPU + 80287 Sockel
- 6 AT + 2 XT Slots
- 8 und **12 MHz** umschaltbar
- 512 KB freier Speicher
- 1 x 1,2 MB/360 KB Laufwerk
- Color- oder Monochr. Grafikkarte (Hercules II komp. 720 x 348 P.)
- Parallele Drucker-Schnittstelle
- Batteriegep. Echtzeituhr/Kalender
- Kapazitive deutsche Normtastatur



Erweiterungen für AT 4 SLC-Serie

2. Laufwerk 360 KB	299,–
20 MB Festplatte mit AT-Controller	889,–
30 MB Festplatte m. RLL AT-Contr.	+ 999,–
Seriell-Karte	79,–
I/0 Plus II Karte	149,–
EGA-Set statt monochr. Karte	+ 1149,–
MS-DOS 3.3 deutsch + GW-Basic	+ 199,–
Professional Multifunktions-Tastatur MK 6000	+ 100,–
9" TTL Monitor grün	+ 150,–
12" Monitor grün od. bern.	+ 229,–
14" TTL Monitor grün, bern./weiß	+ 279,–

PRINTER



MCI *Personal Computer Graphics Printer Plus*

- **voll kompatibel zum** *IBM Personal Computer Graphics Printer*
- **120 Zeichen/sec.**

a. A.



OKI MICROLINE
ML 192 Elite

- **9 Nadel Matrixdrucker**
- **Druckgeschwindigkeit 200 Z./sec.**
- **40 Zeichen/sec. NLQ**
- **Druckpuffer 8 KB**
- **IBM Kompatibel**

a. A.

EGA



Hochauflösender EGA-Monitor

EGA-Karte 249,–

- Auflösung 320 x 200 (CGA Mode) 640 x 350 (EGA Mode)

999,–

TELEFON-HOTLINE-PREISE
MCI Telefonansage (20 sec.)
(0 22 02)

Festplatten & Controller	10 81 40
Grafikkarten	10 81 41
Monitore	10 81 42
Schnittstellenkarten	10 81 43
Drucker	10 81 44
Aktuelle Neuigkeiten	10 81 45



5060 Bergisch Gladbach 2
Bensberger Straße 252
Tel.-Nr.: 02202/1080
Fax: 02202/31009 · Telex: 8873518

12 Monate Garantie auf alle Geräte. Nach der Pang Vo. v. 14. 3. 85 sind wir bei Angeboten gegenüber dem Endverbraucher zur Angabe der **Preise incl. MwSt.** verpflichtet. Für Druckfehler wird nicht gehaftet. **Preise gültig ab 1. 7. 88.** Lieferzeit und Lieferbedingungen auf Anfrage. Änderungen, die technischen Verbesserungen dienen, vorbehalten. Zwischenverkauf vorbehalten.
MCI MICRO COMPUTER INSTRUMENTS GMBH eingetragen AG Bergisch Gladbach · HRB 2575. Herstellung und Vertrieb von Mikrocomputern. 5060 Bergisch Gladbach 2 · Bensberger Straße 252

```
    vo = aktuelle_zeile->vorgaenger;
    next = aktuelle_zeile->nachfolger;
    vo->nachfolger = aktuelle_zeile->nachfolger;
    next->vorgaenger = aktuelle_zeile->vorgaenger;
    aktuelle_zeile = next;
    break;
  }
  cursorposition = 0;
  } /* end delete */

  /* Funktion  : Macro verarbeiten */
  macro()
  {
  int  counter,index,anweisungen;
  long macrozaehler;
  char *mz;
  static char befehl[MACRO_L][ZEILENLAENGE+1];
  LINE *push;

    script_zeiger = script_zeiger->nachfolger;
    macrozaehler = 0;
    /* Scriptzeilen im Block in array befehl speichern */
    for (index=0; index<MACRO_L; index++)
    {
       if (script_zeiger->zeileninhalt[0] == MACRO_STOP)
       {
        mz = script_zeiger->zeileninhalt;
        mz++;
        if ( *mz == GLOBAL )
        {
          macrozaehler = MACRO_SCHLEIFE;
        }
        else
        {
        mz = cut(mz);
        macrozaehler = atol( mz );
        if (macrozaehler>MACRO_SCHLEIFE) macrozaehler=MACRO_SCHLEIFE;
        }
        break;
       }
       if (script_zeiger == script_ende->nachfolger)
       {
         script_zeiger = script_zeiger->vorgaenger;
         break;
       }
       strcpy( befehl[index], script_zeiger->zeileninhalt );
       script_zeiger = script_zeiger->nachfolger;
    }
    /* Befehlsfolgen lt. array befehl zyklisch ausfuehren */
    push = script_zeiger;
    if ( macrozaehler == 0 ) macrozaehler++;
    while( macrozaehler-- > 0 )
    {
       for ( anweisungen=0; anweisungen<index; anweisungen++ )
       {
         /* Suchen in der Anweisungstabelle */
         counter = 0;
         while( TRUE )
         {
            if ( tabelle[counter].funktion == NULL ) break;
            if (befehl[anweisungen][0] == MACRO_START) break;
            if (tabelle[counter].character==befehl[anweisungen][0])
            {
               /* Entsprechende Funktion ausfuehren */
               makro_flag = OKAY;
               strcpy(script_zeiger->zeileninhalt,befehl[anweisungen]);
               ( *tabelle[counter].funktion )();
               break;
            }
            counter++;
         }
         if (makro_flag == FEHLER)
         {  /* Abbruch des Makros */
            anweisungen = index; macrozaehler = 0;
         }
       }
    }
    script_zeiger = push;
  return;
  } /* end macro */

  killchar()     /* Funktion  : Zeichen loeschen */
  {
  unsigned int l, cnt;
  char *src, *dest;

    dest = &aktuelle_zeile->zeileninhalt[cursorposition];
    src  = dest + 1;
    l    = strlen( aktuelle_zeile->zeileninhalt );
    cnt  = l - cursorposition;
    if ( l != 0 ) memcpy( dest, src, cnt );
  return;
  }   /* end killchar */

  /* Funktion : Cursor positionieren */
  cursor_pos()
  {
    getparam( script_zeiger->zeileninhalt );
    switch( *par_1 )
    {
      case CURSOR_D :
                 if ( aktuelle_zeile == text_ende )
                      makro_flag = FEHLER;
                 else
                      aktuelle_zeile = aktuelle_zeile->nachfolger;
                 break;
      case CURSOR_U :
                 if ( aktuelle_zeile == text_anfang )
                      makro_flag = FEHLER;
                 else
                      aktuelle_zeile = aktuelle_zeile->vorgaenger;
                 break;
      case CURSOR_R :
                 if (cursorposition > strlen(aktuelle_zeile->zeileninhalt))
                      makro_flag = FEHLER;
                 else
                      cursorposition++;
                 break;
      case CURSOR_L :
                 if ( cursorposition == 0 )
                      makro_flag = FEHLER;
                 else
                      cursorposition--;
                 break;
      case CAR_RET :
                 cursorposition = 0;
                 break;
      case LINE_END :
                 cursorposition = strlen(aktuelle_zeile->zeileninhalt);
                 break;
      case HOME :
                 aktuelle_zeile = text_anfang;
                 cursorposition = 0;
                 break;
      case END :
                 aktuelle_zeile = text_ende;
                 cursorposition = 0;
    }
      if (cursorposition > strlen(aktuelle_zeile->zeileninhalt))
          cursorposition = strlen(aktuelle_zeile->zeileninhalt);
    return;
  } /* end cursor_pos */

  /* Funktion : Zeile drucken */
  print_line()
  {
  LINE *push;
  char *flags;
  push = aktuelle_zeile;
   getparam( script_zeiger->zeileninhalt );
   flags = par_1; flags = cut( flags );
   while( aktuelle_zeile != script_anfang )
   {
   if (strchr(flags,LINE_NR) != NULL) print_linenumber();
   printf("%s\n",aktuelle_zeile->zeileninhalt);
   if (strchr(flags,GLOBAL) == NULL) break;
   aktuelle_zeile = aktuelle_zeile->nachfolger;
   }
  aktuelle_zeile = push;
  return;
  }  /* end print_line */
```

```
/* Allgemeine Funktionen                                    */
/* Funktion  : File lesen */
fread()
{
int ergebnis;

  file_pointer = fopen(file_name,"r");
  if (file_pointer == NULL)
  {
    printf("\nFile %s existiert nicht !\n",file_name);
    ergebnis = FEHLER;
  }
  else
  {
    /* Text einlesen */
    text_anfang = rdfile(file_pointer,NULL);
    aktuelle_zeile = text_anfang;
    fclose(file_pointer);
    if (overflow)
    {
      printf("\nSpeicher-Überlauf !\n");
      ergebnis = FEHLER;
    }
    else ergebnis = OKAY;
  }
return( ergebnis );
}  /* end fread */

/* Funktion  : File einlesen und im Textspeicher halten */
LINE *rdfile(file,last)
FILE *file;
LINE *last;
{
LINE *anfang,*adresse,*tend,*next;
int  l,i,m;
char character, puffer[ZEILENLAENGE+1];

  anfang = NULL;
  tend = text_ende;
  if (last == text_ende) tend = NULL;
  next = NULL;
  if (last != NULL) next = last->nachfolger;
  /* Text zeichenweise einlesen */
  while ( TRUE )
  {
    for (i=0; i < ZEILENLAENGE+1; i++)
    {
      character = m = fgetc(file);
      if (m == EOF) break;
      if (m == CR)
      {
        i--; continue;
      }
      if (m == LF)  break;
      puffer[i] = character;
    }
    puffer[i] = 0;
    adresse = (LINE *)malloc(sizeof(LINE)+1);
    if (adresse == NULL)
    {
      overflow = 1;   /* Textueberlauf eingetreten */
      break;
    }
    if (anfang == NULL) anfang = adresse;
    /* Vorgaengereintrag */
    adresse->vorgaenger = last;
    /* Nachfolgereintrag */
    adresse->nachfolger = next;
    /* Nachfolgereintrag im Vorgaenger */
    if (last != NULL) last->nachfolger = adresse;
    /* Textanfang setzen */
    else text_anfang = adresse;
    /* Vorgaengereintrag im Nachfolger */
    if (next != NULL) next->vorgaenger = adresse;
    last = adresse;
    /* Zeileninhalt einfuellen */
    strcpy(adresse->zeileninhalt,puffer);
    if (m == EOF) break;
  }
    /* Muss Textende  gesetzt werden ? */
    if (tend == NULL) text_ende =  adresse;
    if (tend == NULL) text_ende = last;
    return(anfang);
  }  /* end rdfile */

/* Funktion  : Script einlesen */
rdscript()
{
int  ergebnis,flag;
LINE *adresse;

ergebnis = OKAY; /* Voreinstellung */
flag = 0;

while( TRUE )
{
    /* Script einlesen */
    adresse = (LINE *)malloc(sizeof(LINE)+1);
    if (adresse == NULL)
    {
        overflow = 1;
        break;
    }
    adresse->vorgaenger = text_ende;
    text_ende->nachfolger = adresse;
    text_ende = adresse;
    if ( flag == 0 )
    {
        flag = 1;
        script_anfang = adresse;
    }
    if (gets( adresse->zeileninhalt ) == NULL) break;
    if (strcmp(adresse->zeileninhalt,"") == NULL) break;
}
if (overflow)
{
  printf("\nSpeicher-Überlauf !\n");
  ergebnis = FEHLER;
}
return( ergebnis );
} /* end rdscript */

/* Funktion  : Text lt. Script umwandeln */
filter()
{
int  counter;

script_zeiger = script_anfang;

   while( TRUE )
   {
      /* Suchen in der Anweisungstabelle */
      counter = 0;
      while( TRUE )
      {
         if ( tabelle[counter].funktion == NULL ) break;
         if ( script_zeiger->zeileninhalt[0] == KOMMENTAR ) break;
         if ( script_zeiger->zeileninhalt[0] == BLANK ) break;
         if ( tabelle[counter].character == script_zeiger->zeileninhalt[0] )
         {
            /* Entsprechende Funktion ausfuehren */
            ( *tabelle[counter].funktion )();
            break;
         }
         counter++;
      }
      script_zeiger = script_zeiger->nachfolger;
      if ( script_zeiger == script_ende->nachfolger ) break;
   }
return;
} /* end filter */

/* Funktion  : Parameter aus der Scriptzeile entnehmen */
getparam( zeiger )
char *zeiger;
{
  while( TRUE)
   {
```

```
      /* 1. Parameter aus Scriptzeile entnehmen */
      par_1 = strchr(zeiger,TRENNZEICHEN);
      if (par_1 == NULL)
      {
        par_1 = par_2 = par_3 = zeiger+strlen(zeiger);
        break;
      }
      par_1++;
      /* 2. Parameter aus Scriptzeile entnehmen */
      par_2 = strchr(par_1,TRENNZEICHEN);
      if (par_2 == NULL)
      {
        par_2 = par_3 = par_1 + strlen( par_1 );
        break;
      }


      *par_2++ = 0;
      /* 3. Parameter aus Scriptzeile entnehmen */
      par_3 = strchr(par_2,TRENNZEICHEN);
      if (par_3 == NULL)
        par_3 = par_2+strlen(par_2);
      else
        *par_3++ = 0;
      break;
    }
return;
} /* end getparam */

/* Funktion  : String von Blanks und Kommentar befreien */
char *cut( string )
char *string;
{
char  *zeiger;
      while( *string == BLANK ) string++;
      zeiger = string;
      while(*zeiger!=0 && *zeiger!=KOMMENTAR && *zeiger!=BLANK) zeiger++;
      *zeiger = 0;
return( string );
} /* end cut */

/* Funktion  : String suchen */
suchen( suchwort )
char *suchwort;
{
int  position,i,found;
char *hilfs_zeiger;
LINE *such_zeile;

  found = FEHLER;  /* Grundeinstellung */
  such_zeile = aktuelle_zeile;
  hilfs_zeiger = aktuelle_zeile->zeileninhalt + cursorposition;
  position = cursorposition;

  while ( such_zeile != text_ende )
  {
    while (*hilfs_zeiger != 0)
    {
      if (strncmp(hilfs_zeiger++,suchwort,strlen(suchwort)) == 0)
      {
        found = OKAY;
        break;
      }
      position++;
    }
    if (found == OKAY) break;
    such_zeile = such_zeile->nachfolger;
    hilfs_zeiger = such_zeile->zeileninhalt;
    position = 0;
  }
  if (found == OKAY)
  {
    aktuelle_zeile = such_zeile;
    cursorposition = position;
  }
  else makro_flag = FEHLER;
return( found );
} /* end suchen */
```

```
/* Funktion  : Wort an Cursorposition gegen Tauschwort ersetzen */
tausche( suchwort, tauschwort )
char *suchwort, *tauschwort;
{
int  n,m;
char c1,c2,*wort;

  /* Suchwort loeschen */
  wort = suchwort;
  while (*wort++ != 0) killchar();

  /* Tauschwort einfuegen */
  wort = tauschwort;
  m    = cursorposition;
  while (*wort != 0)
  {
    /* Einzelnes Zeichen vor Cursorpos. einfuegen */
    c1 = *wort++;
    n  = m++;
    while ( TRUE )
    {
      c2 = aktuelle_zeile->zeileninhalt[n];
      aktuelle_zeile->zeileninhalt[n++] = c1;
      if (c1 == 0) break;
      c1 = c2;
    }
  }
return;
}   /* end tausche */

/* Funktion : Zeilennummer ausgeben */
print_linenumber()
{
int  zaehler;
LINE *zeiger;
   zeiger  = text_anfang;
   zaehler = 1;
   while ( zeiger != aktuelle_zeile )
   {
        zeiger = zeiger->nachfolger;
        zaehler++;
   }
   printf( "%d ", zaehler );
return;
} /* end print_linenumber */

/* Funktion : Helpfile lesen und ausgeben */
rdhelp()
{
int  i;
char line[ZEILENLAENGE+1], *result;
FILE *stream;

  if ((stream = fopen( HELPFILE, "r")) == NULL)
  {
    printf("\nKein Help-File vorhanden !\n" );
  }
  else
  {
    while (1)
    {
        for (i=0; i<23; i++)
        {
          result = fgets(line,ZEILENLAENGE+1,stream);
          if (result == NULL) break;
          printf( "%s", line );
        }
        if (result == NULL) break;
        printf( "\nBeliebige Taste betätigen");
        getch(); printf( "\n\n" );
    } /* end while */
    fcloseall();
  }
} /* end rdhelp */
```

TROST-DATENTECHNIK GMBH

Technologie, Qualität, Service und mehr . . .







DATASTAR 3998.–
PORTABLE-286
- 11"-LCD-Display (640 × 400 Punkte Auflösung) und LCD-Grafik-Adapter
- 80286-CPU
- 6/8/10/12 MHz
- 512 K RAM
- 1,2-MB-NEC-Laufwerk
- 20-MB-Seagate-ST225-Festplatte
- Seriell-/Parallel-Karte
- deutsche Tastatur
- 110/220-V-Autoswitch-Netzteil

DATASTAR-16 1620.–
TURBO PC/XT ohne Monitor
100 % IBM-PC/XT-kompatibel
- 8088-CPU
- 4,77/10 MHz
- 640 K RAM
- „Three in One"-Grundplatine mit integrierten Interface-Karten: Monochrome-Grafik/Color-Grafik-Adapter, 1× Seriell-/1× Parallel-Port, Timer
- 1 Laufwerk 360 K
- deutsche Tastatur mit separatem Cursorblock
- 150 W, Schaltnetzteil

DATASTAR 2934.–
TURBO AT-286 ohne Monitor
100 % IBM-PC/AT-kompatibel
- 80286-CPU
- 6/8/10/12 MHz Geschwindigkeit
- AWARD-BIOS mit Set-up-Programm im ROM
- 512 K RAM, erweiterbar auf 1 MB
- Monochrome-Grafik/Drucker-Karte
- Seriell-/Parallel-Karte
- Western-Digital WD-1003WA2 Floppy-/Harddisk-Controller
- 1,2-MB-NEC-Floppylaufwerk
- 20-MB-Seagate-ST225-Festplatte
- deutsche Tastatur mit separatem Cursorblock
- 200 W, Schaltnetzteil

14"-Monitor 219.–
bernstein, TTL

14"-Monitor 275.–
flat screen, TTL
bernstein oder schwarz/weiß

E.G.A.-Farbmonitor 798.–

P.E.G.A.-Grafik-Karte 398.–

VGA-Karte 598.–

DATASTAR-386 8515.–
TOWER ohne Monitor
- 80386-CPU
- 16/20 MHz
- 2-MB-RAM
- Monochrome-Grafik/Drucker-Karte
- Seriell-/Parallel-Karte
- Western-Digital WD-1003WA2 Floppy-/Harddisk-Controller
- 1,2-MB-NEC-Floppylaufwerk
- 40-MB-Seagate-ST251-Festplatte
- deutsche Tastatur mit separatem Cursorblock
- 200 W, Schaltnetzteil

IBM PC/XT und AT sind eingetragene Warenzeichen von IBM Corp.



Sofort kostenloses Prospektmaterial anfordern
Händleranfragen erwünscht
10 Tage Rückgaberecht ohne Begründung

Trost-Datentechnik GmbH

Ungelsheimer Weg 3, Postfach 30 09 04
4000 Düsseldorf 30
Telefon (02 11) 4 18 58-0, Telefax 02 11/4 18 58-20

Tabelle 1: Editieranweisungen

Anweisungen stehen immer am Anfang der Zeile. Parameter und Optionen werden mit „/" getrennt, beide brauchen von den Anweisungen nicht durch Zwischenräume (Blanks/Tabulator) getrennt zu werden.
Parameter müssen, Optionen können angegeben werden (Optionen in beliebiger Reihenfolge ohne Blanks/Tabulator). Alle Anweisungen wirken ab der aktuellen Cursorposition bzw. der Zeile, in der sich der Cursor befindet. „;" ist Kommentarzeichen.

Funktion:		Cursor positionieren
Syntax:	c	/parameter
Parameter:		+ (Cursor rechts)
		– (Cursor links)
		∧ (Cursor up)
		∨ (Cursor down)
		< (Cursor auf Zeilenanfang)
		> (Cursor hinter Z.-ende)
		h (Cursor auf Textanfang)
		e (Cursor auf Textende)
Optionen:	–	keine

Anmerkung: Ist die neue Cursorposition nach Cursor up/down größer als die ursprüngliche Zeilenlänge, wird der Cursor hinter das letzte Zeichen positioniert.

Funktion:		Zeile löschen
Syntax:	d	
Optionen:	–	keine

Funktion:		String ab Cursorpos. suchen
Syntax:	f	/string/option
Optionen:		p Zeile wird ausgedruckt, wenn der String gefunden wurde
		g Suchen bis File-Ende
		d Zeile wird gelöscht, wenn der String gefunden wurde
		+ Wenn String gefunden wurde, Cursorposition inkrementieren

Funktion:		Zeichen in Cursorposition löschen
Syntax:	k	
Optionen:	–	keine

Funktion:		Zeile drucken
Syntax:	p	/option
Optionen:		g Drucken ab aktueller Zeile
		n Zeilennummer ausgeben

Anmerkung: Zeilennumerierung beginnt bei „1"!

Funktion:		String suchen und ersetzen
Syntax:	s	/suchwort/tauschwort/option
Optionen:		p Zeile wird ausgedruckt, wenn das Suchwort gefunden wurde
		g Suchen/Ersetzen bis File-Ende

Funktion:		Makro definieren
Syntax:	{	(Makroanfang = Alt 123)
		h (Anweisungsliste)
		p
		} option (Makroende = Alt 125)
Optionen:		g Ab Cursorposition bis File-Ende. Zähler (Gibt die Zahl der Durchl. an)

über sinnvolle Bereiche hinaus positioniert wurde, also rechts hinter das Zeilenende, links vor den Zeilenanfang, über den Textanfang oder hinter das Textende.
Hinter dem Makro-Endezeichen kann ein Zählerwert oder die Option „g" (= global, d. h. über die gesamte Datei) folgen. Die maximale Zeilenanzahl bzw. die höchste zulässige Anzahl der Makro-Durchläufe ist aus Sicherheitsgründen begrenzt (→ Datei 'defines.h'). Aus dem Aufbau des Programmes ergibt sich, daß Makros (wegen der zu hohen Stackbelastung) nicht geschachtelt werden dürfen.
Das entsprechend geänderte Script zeigt *Tabelle 4*: Das Makro soll theoretisch 123 mal die definierte Funktion ausführen, wird aber praktisch nach dem fünften Versuch abbrechen.
Für ein zweites, etwas umfangreicheres Beispiel nehmen wir nun die Datei „defines.h" (*Bild 3*). Um diesen Quelltext assembler-kompatibel zu gestalten, wird das Script aus *Tabelle 5* benutzt.
Die Kommandozeile lautet :

```
sed defines.h <script >newfile
```

In „newfile" steht das Ergebnis entsprechend *Tabelle 6.*

Hinweise zur Script-Entwicklung

Aus den vorhandenen Anweisungen lassen sich auch leicht andere zusammensetzen :

- Suche 'string' und positioniere den Cursor dahinter
```
f /string/
f / /
```
- Suche 'string' und hänge 'string_1' an
```
f /string/
s /g/gstring_1/
```
- Lösche 'string'
```
s /string//
```
- Gehe nach Zeilennummer x
```
c /h
{
c /v
}x-1
```

Tabelle 2: Ein erstes Script

```
f   /getparam( /  ; Suche „getparam( "
c   /<            ; Gehe nach Zeilenanfang
c   /∧            ; Gehe eine Zeile zurück
p                 ; Drucke Zeile
c   /∨            ; Gehe eine Zeile vor
p   /n            ; Drucke Zeile mit Nummer
c   /∨            ; Gehe eine Zeile vor
p                 ; Drucke Zeile
```

Tabelle 3: Die Ausgabe des Rechners

```
/* Parameter aus Scriptzeile entnehmen */
83 getparam( script_zeiger->Zeileninhalt );
Suchwort = par_1;
tauschwort = par_2; flags = par_3;
```

Tabelle 4: Eine Lösung mit „Makro"

```
{                 ; Makro-Start
f   /getparam( /  ; Suche "getparam( "
c   /<            ; Gehe nach Zeilenanfang
c   /∧            ; Gehe eine Zeile zurück
p                 ; Drucke Zeile
c   /∨            ; Gehe eine Zeile vor
p   /n            ; Drucke Zeile mit Nummer
c   /∨            ; Gehe eine Zeile vor
p                 ; Drucke Zeile
}123              ; Makro-Ende-Zähler
```

Tabelle 5: Mit diesem Script wird „defines.h" assembler-kompatibel

```
{
f   /*/            ; Finde Kommentarfang
c   /-
k                  ; Lösche "/"
s   /*/;/          ; Neues Kommentarzeichen
f   /*             ; Finde Kommentarende
c   /-
c   /-
k                  ; Lösche Kommentarende
k
}g                 ; Makro für ganzes File
c   /h             ; Cursor auf Textanfang
{
s   /#define / /   ; Lösche String
s   / / EQU/       ; Setze String ein
}g
c   /h
{
c   /∨             ; Cursor positionieren
}13                ; auf Zeile 14
{
p                  ; Drucke 8 Zeilen
c   /∨
}8
```

Tabelle 6: So sieht „defines.h" nach der Bearbeitung aus

```
; Befehle im Script-File
DELETE         EQU   'd'   ; Zeile löschen
FIND           EQU   'f'   ; String suchen
SUBSTITUTE     EQU   's'   ; Stringtausch
MACRO_START    EQU   '{'   ; Macro-Anfang Alt (123)

MACRO_STOP     EQU   '}'   ; Macro-Ende Alt (125)
CURSOR         EQU   'c'   ; Cursorpositionierung
KILLCHAR       EQU   'k'   ; Ein Zeichen löschen
```

```
/* Dateiname : defines.h                                          */
/* Funktion  : Definition der Konstanten                          */

#define HELPFILE        "sed.hlp"
#define TRUE            1
#define OKAY            0
#define FEHLER          1

/* Konstanten fuer Script-File */
#define ZEILENLAENGE    200
#define TRENNZEICHEN    '/'
#define KOMMENTAR       ';'

/* Befehle im Script-File       */
#define DELETE          'd'     /* Zeile loeschen                 */
#define FIND            'f'     /* String suchen                  */
#define SUBSTITUTE      's'     /* Stringtausch                   */
#define MACRO_START     '{'     /* Macro beginnt mit Alt(123)     */
#define MACRO_STOP      '}'     /* und endet mit Alt(125)         */
#define CURSOR          'c'     /* Cursorpositionierung           */
#define KILLCHAR        'k'     /* Einzelnes Zeichen loeschen     */

/* Flags fuer Befehle           */
#define GLOBAL          'g'
#define PRINT           'p'
#define LINE_NR         'n'     /* Zeilennummer ausgeben          */
#define CURSOR_R        '+'     /* Cursor 1 nach rechts           */
#define CURSOR_L        '-'     /* Cursor 1 nach links            */
#define CURSOR_U        '^'     /* Cursor 1 nach oben             */
#define CURSOR_D        'v'     /* Cursor 1 nach unten            */
#define CAR_RET         '<'     /* Cursor auf Zeilenanfang        */
#define LINE_END        '>'     /* Cursor auf Zeilenende          */
#define HOME            'h'     /* Cursor auf Textanfang          */
#define END             'e'     /* Cursor auf Textende            */

/* Konstanten fuer Macroverarb. */
#define MACRO_L         30      /* Max. Laenge (Zeilen) eines Macros */
#define MACRO_SCHLEIFE  10000   /* Max. Anzahl der Macrodurchlaeufe  */

#define BACK_SPACE      8
#define CR              13
#define LF              10
#define BLANK           32

/***** ENDE *****/
```

Bild 3. Alle Konstanten-Deklarationen sind in dieser Datei unter dem Namen DEFINES.H zusammengefaßt

```
/* Dateiname : befehle.c                                  */
/* Auflistung der Script-Funktionen                       */
int substitute(); /* String ersetzen                      */
int finden();     /* String suchen                        */
int delete();     /* Zeile loeschen                       */
int macro();      /* Macro verarbeiten                    */
int cursor_pos(); /* Zeigerpositionierung                 */
int killchar();   /* Zeichen loeschen                     */
int print_line(); /* Aktuelle Zeile drucken               */

/* Anweisungstabelle mit je 2 Eintraegen :                */
/* 1. Anweisung                                           */
/* 2. Adresse auszufuehrende Funktion                     */

struct anweisung tabelle[] =
{
        { SUBSTITUTE,  substitute  },
        { FIND      ,  finden      },
        { DELETE    ,  delete      },
        { MACRO_START, macro       },
        { CURSOR    ,  cursor_pos  },
        { KILLCHAR  ,  killchar    },
        { PRINT     ,  print_line  },
/*          Endemarkierung              */
        { 0         ,  NULL        }
};
```

Bild 4. Die Datei „Befehle.C" enthält alle Befehle und muß bei Erweiterungen des Programmes angepaßt werden

Tabelle 7: Mit dieser Batch-Datei lassen sich die Quelltexte aus Bild 1...4 zu einem Programm übersetzen und binden

```
echo off
msc sed /AL;
if errorlevel 1 goto mscfe
link sed;
if errorlevel 1 goto linkfe
goto ende
:mscfe
echo Compiler-Fehler!
goto ende
:linkfe
echo Link-Fehler!
:Ende
del sed.obj
```

Ist ein Makro mit der Option „g" (global) oder einem Zählerwert abgeschlossen, muß in den meisten Fällen ein Cursor-Home-Kommando „c /h" folgen. Der fiktive Cursor steht immer an der Position, an der die letzte Anweisung erfolgreich durchgeführt wurde.

Für einen Test ist es sinnvoll, hinter jeder Such- und Ersetzoperation die Option „p" (print) anzugeben, um die Entwicklung der neuen Datei verfolgen zu können.

Es ist möglich, Textdateien zu manipulieren, die z. B. mit WordStar im Document-Modus erstellt wurden, wenn auch das Script im Document-Modus geschrieben ist. Wichtig: Es muß unbedingt darauf geachtet werden, daß für die alte und neue Datei nicht die gleichen Namen verwendet werden. Es ist in diesem Fall die Eigenschaft von MS-DOS, die Standardausgabe (d. h. die neue Datei) vor dem Programmdurchlauf zu löschen!

Erweiterungen

Wer schon viel mit Editoren gearbeitet hat, dem wird auffallen, daß sich mit dem Stream-Editor nicht alle Funktionen eines umfangreichen Textsystems ersetzen lassen. Darum folgen hier noch Tips für die mögliche Erweiterung des Programmes:

Das Programm besteht aus 4 Dateien:

- sed.c Hauptprogramm und Unterfunktionen
- defines.h Definition der Konstanten
- befehle.c Befehlstabelle
- sed.hlp Hilfsdatei

Erweiterungen sollten grundsätzlich zum Hauptprogramm hinzugebunden ('gelinkt') werden. Die Datei „defines.h" enthält alle nötigen Konstanten-Deklarationen und wird in die Erweiterungen per „#include"-Anweisung eingebunden.

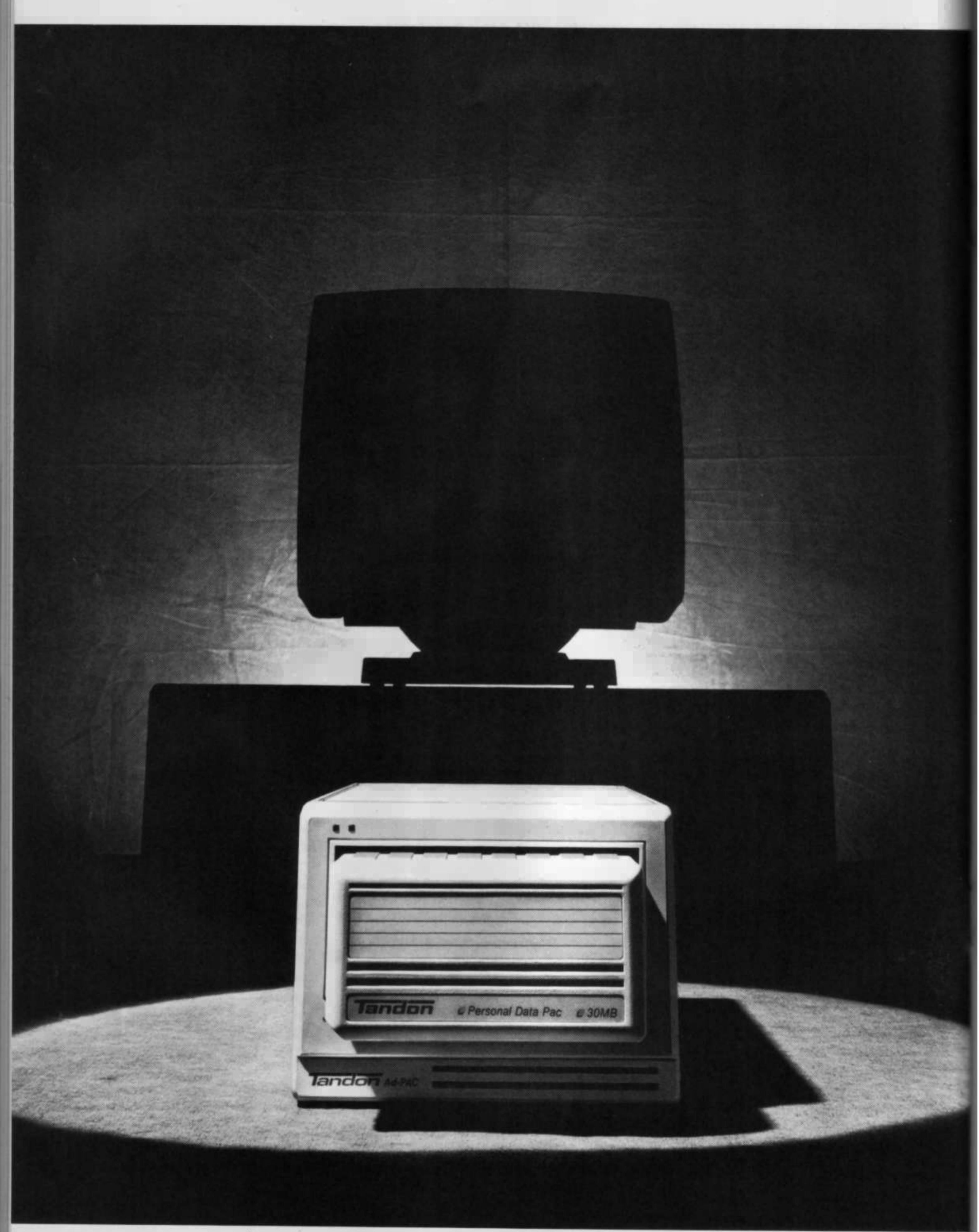
Die Datei „befehle.c" *(Bild 4)* enthält eine Tabelle, in der die Editierzeichen und die dazugehörigen Funktionen aufgelistet sind. Das Hauptprogramm greift auf diese Tabelle zu. Laut Script wird die entsprechende Funktion ausgeführt. Für die Implementierung zusätzlicher Anweisungen muß diese Tabelle erweitert werden.

Das Programm sed.c einschließlich Erweiterung sollte unbedingt mit dem Large-Model compiliert (Option /AL) werden, da sonst der Speicherbereich für Programm und Text auf je 64 KByte beschränkt bleibt! Für den Compiler/Linker-Durchlauf hat sich die Batch-Datei „compile.bat" in *Tabelle 7* als sehr nützlich erwiesen

Literatur:

[1] IBM-DOS-Handbuch, IBM

[2] *Edgar Huckert:* C-Toolbox Textverarbeitung, Verlag Markt und Technik, Haar bei München, 1986



Data-Pac-Technologie von Tandon revolutioniert den PC-Markt.

Endlich Frieden. Wer jetzt mit dem Add-on-PAC von Tandon aufrüstet, der muß nie wieder nachrüsten.

Mit dem Add-on-PAC findet Ihr kompatibler PC umstandslos Anschluß an Tandons bahnbrechende Technologie der mobilen Festplatten: Er öffnet Ihr System für endlos viele Data Pacs und damit für unbegrenzte Speicherkapazität.

Weitere grenzüberschreitende Eigenschaften der Aufrüstung mit dem Add-on-PAC: Data Pacs sind grenzenlos transportabel, grenzenlos versendbar und fast grenzenlos sicher, weil Sie sie einfach wegschließen können. Sie sind so handlich wie ein kleines Buch und mindestens ebenso robust: kein



Buchstabe, keine Zahl fällt raus, wenn ein Data Pac mal runterfällt.

Es gibt nur eine Sache auf der ganzen PC-Welt, die noch fortschrittlicher ist als die Aufrüstung mit dem Add-on-PAC, und das ist ein Original-Data-Pac-PC von Tandon, wie z. B. der PAC 286 oder der Tandon 386. Da wir Sie hier aber nur auf die Möglichkeit der friedlichen Koexistenz Ihres vorhandenen Systems mit Tandons revolutionärer Neuerung hinweisen wollen, bitten wir Sie, uns anzurufen oder uns zu schreiben, wenn Sie Näheres über die Data-Pac-Technologie wissen möchten:

Tandon Computer GmbH, Wächtersbacher Straße 59–61, 6000 Frankfurt am Main 61, Tel.: 069-42095-201 oder 202.

Tandon

Wir haben mehr im Kasten.

Armin Giesel

DFÜ-Hilfe für den Mac

Viele Terminalprogramme, wie zum Beispiel MacTerminal, haben einen Schwachpunkt: Man kann keine Macros anlegen, die einem das Einstellen der Modems und das Wählen einer Nummer abnehmen. Diese Lücke füllt das hier beschriebene Desk-Accessory – jederzeit unter dem Apfel verfügbar.

Das Programm wurde ursprünglich in TML-Pascal geschrieben, ist in der vorliegenden Version aber an MPW-Pascal angepaßt worden. Die notwendigen Änderungen waren nicht von grundlegender Natur, so daß eine Zurückführung auf die TML-Version keine Schwierigkeiten bereiten dürfte.

Allgemeines über Desk Accessorys (DAs)

DAs sind im Prinzip nichts anderes als Treiber (DRVR). Das System erwartet von Ihnen fünf Einsprungadressen für OPEN-, CONTROL-, STATUS-, PRIME- und CLOSE-Aufrufe. Wird ein DA geöffnet, wird die OPEN-Routine gerufen. Diese muß dann die nötige Initialisierung durchführen (Fenster öffnen usw.). CLOSE ist für das Schließen zuständig, alle anderen Aufrufe laufen (bei DAs) über die CONTROL-Routine. Die Parameter sind bei allen Routinen gleich: Ein Zeiger auf einen Block mit Systemparameter und ein Zeiger auf einen zweiten Block diverser Variablen. Vom Parameterblock ist zunächst einmal die Variable csCode interessant. Für den darin enthaltenen Code sind folgende Konstanten vereinbart:

accEvent
Ein Event ist zu bearbeiten. Die Variable csParam des Parameterblocks enthält einen Zeiger auf den Event.

accRun
Es handelt sich um einen periodischen Aufruf. Die Periodendauer wird bei der Definion der DRVR-Resource in 1/60 Sekunden angegeben (drvrDelay).

accCursor
Dies ist der häufigste CONTROL-Aufruf. Immer wenn gerade nichts anderes zu tun ist, werden geöffnete DAs auf diese Art gerufen, und bekommen so Rechenzeit zugewiesen.

accMenu
Das DA-Menü wurde angewählt. Die Variable csParam des Parameterblocks enthält die Nummer des gewählten Menüpunktes.

15	14	13 11	10 5	4 0
1	1	Typ	Master-ID	Sub-ID

Bild 1. Schema der Systemresourcen

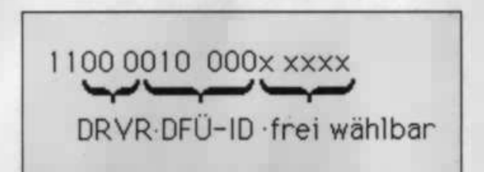


Bild 2. Schema der DFÜ-Hilfe-Resource

accUndo
Menüpunkt 'widerrufen' des 'Bearbeiten'-Menüs wurde gewählt. Das 'Bearbeiten'-Menü gehört (neben Apfel- und Ablagemenü) zu den standardisierten Menüs, und sollte in jedem Programm gleich sein. Auf diese Weise kann es von DAs mitbenutzt werden.

accCut
Menüpunkt 'ausschneiden' des 'Bearbeiten'-Menüs... (s.o.)

accCopy
Menüpunkt 'kopieren'... (s.o.)

accPaste
Menüpunkt 'einsetzen'...(s.o.)

accCut
Menüpunkt'löschen'...(s.o.)

Von den diversen Variablen im zweiten Block ist zum einen die Variable DctlStorage von Bedeutung. Da für ein DA keine globalen Variablen deklariert werden können, muß im Heap dafür Platz reserviert werden. Den Zeiger (besser: Handle) auf diesen Speicherplatz trägt man in dieser Variable ein, womit er für jede Routine verfügbar ist. Falls das DA einen eigenen Menüpunkt besitzt (wie z. B. Tastatur-DA), muß die ID-Nummer dieses Menüs in die Variable DctlMenu eingetragen werden. Diese Nummer muß allerdings zuerst neu errechnet werden, denn der Font/DaMover verändert u. U. die Resource-IDs beim Einbinden ins System (s.u..)! Die DFÜ-Hilfe besitzt ein Menü, das nur einen Menüpunkt (Info...) enthält. Bei Anwahl desselben erscheint lediglich in einem separaten Fenster eine Copyright-Notiz. Es existiert nur der Vollständigkeit halber und kann daher ohne Funktionsbeeinträchtigung auch weggelassen werden. Schließlich wird in DctlWindow noch der Pointer für das DA-Fenster eingetragen. Auch hier gilt neuerrechnen (s.u.)!

Die Resource-IDs von DAs

Bekanntlich darf in einer Resource-Datei eine ID-Nummer eines Resourcetyps nur einmal vorkommen. Bei Programmen liegt die Vergabe der ID-Nummern alleine beim Programmierer, der auf die Einhaltung dieser Bedingung zu achten hat. DAs aber werden, damit sie unter dem Apfelmenü zur Verfügung stehen, mit dem Font/DA-Mover in die Systemdatei eingebunden. Und auch hier gilt die Bedingung: Keine doppelten Resource-IDs! Damit taucht schon der erste Konflikt auf, denn woher soll der Programmierer eines DAs nun wissen, welche IDs noch frei sind? Aus diesem Grund hat man sich für alle Resourcen der Sgstemdatei ein besonderes Nummernsystem ausgedacht. *Bild 1* zeigt, wie die einzelnen Bits einer solchen System-Resource-ID zugeordnet sind: Die Bits 15 und 14 sind immer auf eins. Die Bits 13...11 enthalten einen Code, der angibt, zu welcher Art die betreffende Resource gehört. Bei Treibern (DRVR) sind diese Bits alle auf Null. Die Bits 10...5 enthalten die Resourcenummer der „Master-"Resource. Die Master-Resource ist diejenige Resource, zu der die Resource mit der vorliegenden ID-Nummer gehört. Beispiel DFÜ-Hilfe: Master-Resource ist der Treiber (DRVR), der den Programmcode enthält. Für ihn wurde als Resource-ID (willkürlich) die Zahl 16 definiert. Damit müssen alle (Sub-)Resourcen,

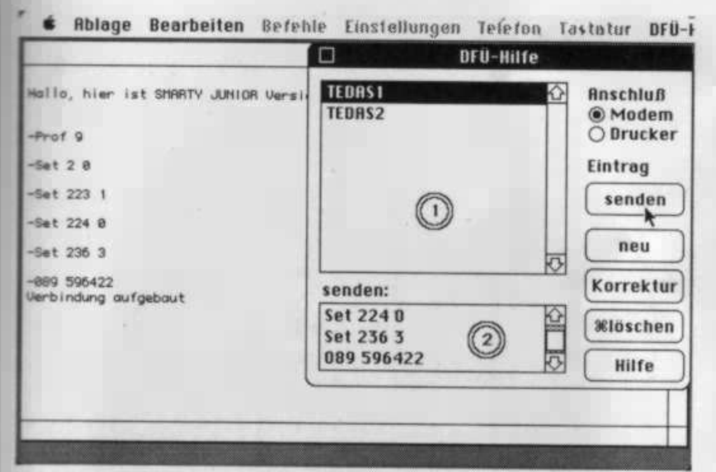


Bild 3. DFÜ-Hilfe in Aktion

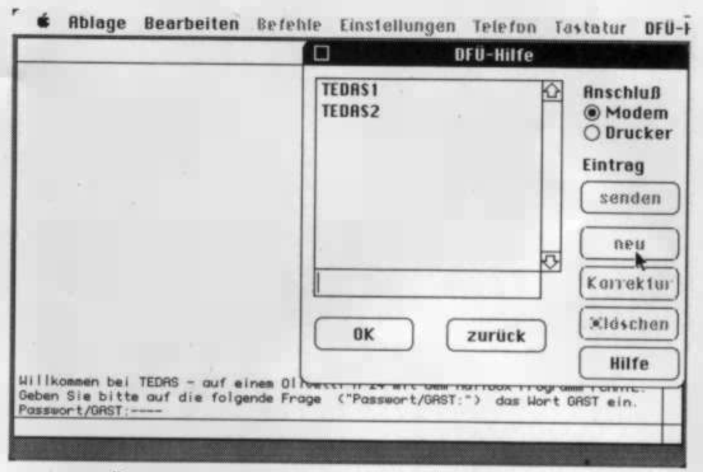


Bild 4. DFÜ-Hilfe nach Anklicken des 'NEU'-Knopfes

die zur DFÜ-Hilfe gehören (Fenster, Dialoge...), in den Bits 10...5 die Zahl 16 enthalten. Die Bits 4...0 schließlich dienen zur Unterscheidung der einzelnen Sub-Resourcen. Ein DA kann demnach maximal 32 Resourcen eines bestimmten Typs besitzen. *Bild 2* zeigt das Bitmuster für die Sub-Resourcen der DFÜ-Hilfe. Es ergibt sich ein Bereich von (Hex) $C200 (für x=0) bis $ C21F (für x=1) oder dezimal −15872...−15841. Demnach müssen alle Resource-IDs der DFÜ-Hilfe-Resourcen in diesem Bereich liegen (vgl. Resource-Definitionen). Nun kann aber trotz allem doch noch ein Resource-ID-Konflikt entstehen: Wenn nämlich in der Systemdatei bereits ein DA existiert, für das dieselbe (Master-) Resource-ID vergeben wurde, wie für das neue, dazu kommende DA. In diesem Fall tritt der Font/DA Mover in Aktion. Er ändert einfach die Resource-ID des neuen DAs in eine noch nicht vorhandene ID-Nummer um und korrigiert entsprechend der Vorgabe nach Bild 1 sämtliche Resource-IDs des neuen DAs! Aus diesem Grund können die Resource-IDs von Fenstern, Dialogen etc. bei DAs nicht als Konstanten deklariert werden, sondern müssen vom DA zur Laufzeit nach dem Schema in Bild 1 neu errechnet werden. Dazu muß natürlich die neue, vom Font/DA Mover vergebene Nummer verwendet werden. Diese erhält man durch invertieren der Variablen DctlRefNum. Da in Pascal aber nicht invertiert sondern nur negiert werden kann, erhält man eine etwas „schräg" aussehende Formel zur Berechnung der Resource-ID:

ID = $C000 − 32 * (DctlRefNum+1)
oder
ID = $BFE0 − 32*DctlRefNum

```
"DFÜ-Hilfe"          ƒ   "DFÜ-Hilfe.DRVW" "DFÜ-Hilfe.r"
    Rez -rd -c DMOV -t DFIL "DFÜ-Hilfe.r" -o "DFÜ-Hilfe"
"DFÜ-Hilfe.DRVW"     ƒ   "DFÜ-Hilfe.p.o"
    Link -w -rt DRVW=0 ∂
        -sg DFUE ∂
        "{Libraries}"DRVRRuntime.o  # This must preceed Memory.p.o ∂
        "DFÜ-Hilfe.p.o" ∂
        "{Libraries}"Interface.o ∂
        "{PLibraries}"Paslib.o ∂
        -o "DFÜ-Hilfe.DRVW" -c "????" -t "????"
"DFÜ-Hilfe.p.o"      ƒ   "DFÜ-Hilfe.p"
    Pascal "DFÜ-Hilfe.p"
```

Bild 5. Die 'Make-'Datei zum Übersetzen und Binden unter MPW

Speicherung der Macros

Für kleinere Datenverwaltungsaufgaben, wie etwa bei der hier beschriebenen DFÜ-Hilfe, läßt sich der Resource-Manager wunderbar als Datenbank „mißbrauchen". Als Datenraum fungiert die Resource-Datei 'DFÜ-Hilfe.Data', die, falls noch nicht vorhanden, automatisch im Systemordner angelegt wird. Alle Daten werden als 'STR'-Resource gespeichert. Prinzipiell könnte man natürlich auch jeden anderen Resourcetyp wählen, aber auf diese Weise konnte bei der Entwicklung des DA seine korrekte Funktion problemlos mit jedem Resource-Editor überprüft werden. Identifiziert werden die einzelnen Resourcen hier allerdings nicht anhand ihrer ID-Nummer (sie wird per Zufallsfunktion zugewiesen), sondern anhand ihres Namens. Dieser ist nämlich gleichzeitig das vom Benutzer einzugebende Kennwort, unter dem das Macro gespeichert werden soll. Das Macro selbst wird, wie gesagt, als Stringresource gespeichert. Daß dieses Macro auch mehrzeilig sein kann und damit CR-Zeichen enthält, stört hier nicht weiter. Es muß lediglich beachtet werden, daß die gesamte Sequenz nicht mehr als 255 Zeichen entält; aber das dürfte kein Problem sein (zur Übertragung von größeren Texten ist die DFÜ-Hilfe ja nicht gedacht).

Verwaltung der Macros

Zur Verwaltung der oben beschriebenen 'Resource-Datenbank' wird der List-Manager verwendet. Er beinhaltet Routinen zum Löschen und Einfügen von Einträgen in eine definierte Liste, übernimmt das Aktualisieren, Aktivieren, Desaktivieren und Scrollen der Liste, und er bearbeitet Mausklicks innerhalb der definierten Abmessungen der Liste. Auf einfache Weise kann so festgestellt werden, ob ein Feld der Liste vom Benutzer aktiviert worden ist und, wenn ja, welches Feld aktiviert wurde. Eine Liste ist für den List-Manager ein zweidimensionales Feld gleich großer Zellen. Sie wird definiert über die Größenangabe einer Zelle und die Angaben über die Anzahl von Zeilen und Spalten der Liste (Dimension). Beide Angaben werden der Initialisierungsprozedur als Variable vom Typ RECT übergeben, wobei für die Zellendefinition die Maßeinheit Pixel gilt, bei Dimension dagegen die Zellenanzahl gemeint ist. So führt eine zweispaltige Liste über 9 Zeilen, wobei die erste Spalte und die erste Zeile den Index Null besitzen, zu: 0,0,2,9 (links, oben, rechts, unten). Die für die Definition erforderliche Zeilenzahl, die der Anzahl der momentan gespeicherten Macros entspricht, bekommt man in unserem Fall mit Hilfe der Resourcemanager-Funktion Count1Resources. Im Gegensatz zu CountResources zählt diese Funktion nur die Resourcen des gewünschten Types in der gerade aktiven Resourcedatei. Damit dies auch die richtige ist, sollte man dies mit

```
/* Resource-Definition für DFÜ-Hilfe */

#include "Types.r"              /* system types */
#include "MPWTypes.r"           /* 'DRVW' type */

type 'DRVR' as 'DRVW';          /* Map 'DRVW' => 'DRVR' */

#define DriverID    16

/* Resource ID's für Resourcen 'OWNED' von Desk Accessory's
 *  (System Resourcen von Typ 'DRVR' ):
 *
 * Bits:      15 14 13 12 11     10 9 8 7 6 5        4 3 2 1 0
 * Bedeutung: 1  1  0  0  0      [Resource-Id]       [variable]
 *
 * Daraus ergibt sich für ID-Nr 16 ein Bereich von
 *   -15872   bis   -15857    ($C200 -> $C20F)
 */
#define BasisId -15872

resource 'DRVR' (DriverID, "\0x00DFÜ-Hilfe", purgeable) {
    dontNeedLock,             /* DA ist frei verschiebbar */
    needTime,                 /* periodische Aufrufe nötig (zum senden) */
    dontNeedGoodbye,          /* ??? */
    noStatusEnable,
    ctlEnable,                /* DAs sind nur über Control-Calls erreichbar */
    noWriteEnable,
    noReadEnable,
    60,                       /* drvrDelay - im Sekundenrythmus rufen */
    everyEvent,               /* drvrEMask - alle Events werden bearbeitet */
    0,                        /* drvrMenu  */
    "DFÜ-Hilfe",              /* drvrName  */
    $$resource("DFÜ-Hilfe.DRVW", 'DRVW', 0)
};

#define AllItems    0b1111111111111111111111111111111   /* 31 flags */

Resource 'MENU' (BasisId, "DFÜ-Hilfe", preload)
    {BasisId, TextMenuProc,
     AllItems,
     enabled, "DFÜ-Hilfe",
       {   "Infos…",       noIcon,noKey,noMark,plain}
    };

Resource 'DLOG' (BasisId,"DFÜ-Hilfe")       /* Hauptfenster */
    {   {40, 220, 280, 510},
        rDocProc, inVisible, GoAway, 0, BasisId, "DFÜ-Hilfe"};

Resource 'DLOG' (BasisId+1,"Hilfe1")        /* Hilfefenster 1 */
    {   {40, 10, 300, 425},
        dBoxProc, Visible, GoAway, 0, BasisId+1, "Hilfe1"};

Resource 'DLOG' (BasisId+2,"Hilfe2")        /* Hilfefenster 2 */
    {   {40, 10, 300, 425},
        dBoxProc, Visible, GoAway, 0, BasisId+2, "Hilfe2"};

Resource 'DLOG' (BasisId+3,"Hilfe3")        /* Hilfefenster 3 */
    {   {40, 10, 300, 425},
        dBoxProc, Visible, GoAway, 0, BasisId+3, "Hilfe3"};
```

Bild 6. Der Pascal-Quell-Text des Programmes

```
Resource 'DITL' (BasisId)                    /* für Hauptfenster */
    {{  { 85,210,110,285},  button      { enabled,"senden"};
        {180,210,205,285},  button      { enabled,"\0x11löschen"};
        {120,210,145,285},  button      { enabled,"neu"};
        {180, 12,226,178},  EditText    { enabled,""};
        { 10, 10,154,180},  UserItem    { enabled};      /* Box für Macroliste */
        { 10,180,154,195},  UserItem    { enabled};      /* Scrollbar "  " */
        {157, 12,172,178},  EditText    { enabled,""};
        {190, 10,215, 85},  button      { enabled,"OK"};
        {190,110,215,185},  button      { enabled,"zurück"};
        {210,210,235,286},  button      { enabled,"Hilfe"};
        { 25,210, 40,285},  Radiobutton { enabled,"Modem"};
        { 40,210, 55,285},  Radiobutton { enabled,"Drucker"};
        { 10,210, 25,285},  StaticText  {disabled,"Anschluß"};
        { 65,210, 80,285},  StaticText  {disabled,"Eintrag"};
        {160, 10,175, 85},  StaticText  {disabled,"senden:"};
        {150,210,175,285},  button      { enabled,"Korrektur"};
        {177,180,229,196},  UserItem    { enabled};      /* Box für Macro */
        { 85,210,110,285},  button      { enabled,"abbrechen"}
    }};

Resource 'DITL' (BasisId+1)                 /* für Hilfefenster 1 */
    {{  {235,170,260,245},  button      { enabled,"weiter"};
        {  5,  5, 20,300},  StaticText  {disabled,"Über DFÜ-Hilfe"};
        { 30,  5,130,420},  StaticText  {disabled,"Mit diesem Programm können "
            "Macros unter einem Kennwort gespeichert und wieder abgerufen "
            "werden."};
        {110,  5,200,420},  StaticText  {disabled,"Nach dem Start des Program"
            "mes sehen Sie im großen Ausschnitt eine Liste der bisher "
            "eingetragenen Kennwörter (Macros). Nach dem Anklicken (aktivieren) "
            "eines Kennwortes erscheinen im unteren Ausschnitt das unter diesem "
            "Kennwort gespeicherte Macro."}
    }};

Resource 'DITL' (BasisId+2)                 /* für Hilfefenster 2 */
    {{  {235,170,260,245},  button      { enabled,"weiter"};
        {  5,  5, 20,420},  StaticText  {disabled,
              "So arbeiten Sie mit DFÜ-Hilfe"};
        { 30,  5, 65,420},  StaticText  {disabled,
            "• Benutzen Sie 'Senden', um aktivierte Macros an's Modem\0x0D"
            "    zu schicken\0x0D"};
        { 70,  5,125,420},  StaticText  {disabled,
            "• Benutzen Sie 'Neu', wenn Sie ein neues Macro speichern\0x0D"
            "  wollen. Tragen Sie dazu zunächst das Kennwort und\0x0D"
            "  anschließend den Macrotext ein."};
        {130,  5,190,420},  StaticText  {disabled,
            "• Benutzen Sie 'Korrektur' um ein Kennwort zu ändern, und\0x0D"
            "  'Löschen' um ein Macro zu entfernen.\0x0D"
            "  Aus Sicherheitsgründen muß beim Anwählen von 'Löschen'\0x0D"
            "  die \0x11-Taste gedrückt sein!"};
        {195,  5,230,420},  StaticText  {disabled,
            "• aktivierte Macros editieren Sie in der gewohnten Weise\0x0D"
            "  (wie in einer Textverarbeitung).\0x0D"}
    }};

Resource 'DITL' (BasisId+3)                 /* für Hilfefenster 3 */
    {{  {235,170,260,245},  button      { enabled,"ok"};
        {  5,  5, 20,425},  StaticText  {disabled,"Technische Hinweise"};
        { 30,  5, 90,420},  StaticText  {disabled,
            "Bei der Übertragung der Macros legt 'DFÜ-Hilfe' nach jedem CR "
            "(Carriage Return = Zeilenende) eine Pause von einer Sekunde ein, "
            "um dem Modem Zeit zur Bearbeitung der empfangenen Sequenz zu geben."};
```

```
        {110,  5,155,420},  StaticText  {disabled,
            "DFÜ-Hilfe ist lediglich als Unterstützung gedacht, und übernimmt "
            "daher auch keine Initialisierung der Schnittstelle!"};
        {165,  5,205,420},  StaticText  {disabled,
            "Zur Sicherung Ihrer Daten kopieren sie die im Systemordner "
            "befindliche Datei 'DFÜ-Hilfe.Data' auf Ihre Sicherungsdiskette."}
    }};
            {$D+} { MacsBug symbols on }
            {$R-} { No range checking  }
Unit DFUE;
                        Interface

Uses    {$Load MemTypes.dump}   MemTypes,
        {$Load Quickdraw.dump}  QuickDraw,
        {$Load OSIntf.dump}     OSIntf,
        {$Load ToolIntf.dump}   ToolIntf,
        {$Load PackIntf.dump}   PackIntf;

Function DrvrOpen    (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
Function DrvrControl (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
Function DrvrStatus  (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
Function DrvrPrime   (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
Function DrvrClose   (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;

                    Implementation

Const   DataName    = ':DFÜ-Hilfe.Data'; {Resourcedatei}

        { die verschiedenen vorhandenen Knöpfe und sonstige Bereiche: }
        Sendeknopf      = 1;
        Loeschknopf     = 2;
        NeuKnopf        = 3;
        MacroTextBox    = 4;
        ListenBox       = 5;    { Macroliste }
        ScrollBox       = 6;    { für Macroliste }
        MacroIdBox      = 7;    { Macrobezeichnung (nur für NEU/KORREKTUR) }
        okKnopf         = 8;
        ZurueckKnopf    = 9;
        HilfeKnopf      =10;
        ModemKnopf      =11;
        DruckerKnopf    =12;
        AnschlussText   =13;    { StaticText 'Anschluß' }
        EintragText     =14;    { StaticText 'Eintrag' }
        sendenText      =15;    { StaticText 'senden:' }
        KorrekturKnopf  =16;
        MacroScrollBar  =17;
        AbbruchKnopf    =18;

 TYPE   Globale_Var = Record
                        Dialog      : DialogPtr;     { Hauptfenster }
                        DataRefNr   : Integer;       { RefNr. des 'Datenraumes' }
                        Liste       : ListHandle;    { Macroliste }
                        PortRefNr   : Integer;       { Drucker/Modemport }
                        TextResHdl  : Handle;        { aktiviertes Macro }
                        TextScroll  : ControlHandle; { Scrollbar für Macro }
                        SendeHdl    : Handle;        { zu sendende Daten }
                      end;
        Global_Ptr  = ^Globale_Var;
        Global_Hdl  = ^Global_Ptr;

Procedure DItem_GetText(Dialog: DialogPtr; ItemNr: Integer; var Txt: Str255);
                                        { Item-Text in String kopieren }
    Var     ItemType: Integer;
            ItemHdl : Handle;
            Box : Rect;
    Begin
      GetDItem(Dialog,ItemNr,ItemType,ItemHdl,Box);
      GetIText(ItemHdl,txt);
    end;

Procedure DItem_SetText(Dialog: DialogPtr; ItemNr: Integer; Txt: Str255);
                                        { String als Item-Text einsetzen }
    Var     ItemType: Integer;
            ItemHdl : Handle;
            Box     : Rect;
    Begin
      GetDItem(Dialog,ItemNr,ItemType,ItemHdl,Box);
      SetIText(ItemHdl,txt);
      TECalText(DialogPeek(Dialog)^.TextH);
    end;

Procedure DItem_Hilite(Dialog: DialogPtr; ItemNr: Integer; ein: Boolean);
                                    { Item (Knopf) aktivieren/deaktivieren }
    Var     Status  : Integer;
            ItemType: Integer;
            Item    : Handle;
            Box     : Rect;
    Begin
      GetDItem(Dialog,ItemNr,ItemType,Item,Box);
      If ein then Status := 0
             else Status := 255;
      HiliteControl(Pointer(Item),Status);
    end;

Procedure DItem_SetCtlValue(Dialog: DialogPtr; ItemNr: Integer; Value: Integer);
                                    { CtlValue eines Items setzen}
    Var     ItemType: Integer;
            Item    : Handle;
            Box     : Rect;
    Begin
      GetDItem(Dialog,ItemNr,ItemType,Item,Box);
      SetCtlValue(Pointer(Item),Value);
    end;

Procedure DItem_GetBox(Dialog: DialogPtr; ItemNr: Integer; var Box: Rect);
                                    { Abmessungen eines Items feststellen }
    Var     ItemType    : Integer;
            ItemHdl     : Handle;
    Begin
      GetDItem(Dialog,ItemNr,ItemType,ItemHdl,Box);
    end;

Procedure Deactivate_DA(SysCom: DCtlPtr);
                                    { DA deaktivieren }
    Var Globals : Global_Hdl;

    Begin
      Globals := Pointer(SysCom^.DCtlStorage);
      DeleteMenu(SysCom^.DCtlMenu); { DFÜ-Menüpunkt entfernen }
      DrawMenuBar;
      LActivate(false,Globals^^.Liste); {Liste deaktivieren}
    end;
```

```
Procedure V_Action(Scroll: ControlHandle; PartCode: Integer);
                                { Mausklick im Scrollbereich behandeln }
    Var     aktueller_Wert,
            Maximalwert,
            Schrittweite     : Integer;
    Begin
      aktueller_Wert := GetCtlValue(Scroll);
      Maximalwert := GetCtlMax(Scroll);
      Case PartCode of
        inUpButton  : If aktueller_Wert > 5 then Schrittweite := -5
                                       else Schrittweite := -aktueller_Wert;
        inDownButton: If aktueller_Wert < (Maximalwert-5) then Schrittweite := 5
                                else Schrittweite := Maximalwert-aktueller_Wert;
        inPageUp    : Schrittweite := Maximalwert - aktueller_Wert;
        inPageDown  : Schrittweite := -aktueller_Wert;
        otherwise   Schrittweite := 0;
      end;
      SetCtlValue(Scroll,aktueller_Wert+Schrittweite);
      TEScroll(0,-Schrittweite,DialogPeek(GetCRefCon(Scroll))^.TextH);
    end;
{______________________________________________________________________}

Function DrvrOpen(SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;


VAR     Globals : Global_Hdl;    { globale Daten }

Procedure DialogFenster_oeffnen;

    Var     ResourceID  : Integer;
            HilfsPtr    : Ptr;
            WPeek       : WindowPeek;
    Begin
      ResourceID := $BFE0 - 32*(SysCom^.DCtlRefNum);{ vom Font/DA Mover geändert!}
      SysCom^.DCtlMenu := ResourceID;
        { Platz für globale Daten schaffen: }
      SysCom^.DctlStorage := NewHandle(SizeOf(Globale_Var));
      HilfsPtr := NewPtr($1000); { Window-Record nicht ans HEAP-Ende legen! }
      SysCom^.DctlWindow := Pointer(GetNewDialog(ResourceID,nil,pointer(-1)));
      DisposPtr(HilfsPtr);
      WindowPeek(SysCom^.DCtlWindow)^.WindowKind := SysCom^.DCtlRefNum;
      SetPort(Pointer(SysCom^.DctlWindow));
    end; {DialogFenster_oeffnen}

Procedure Init_Items(Dialog: DialogPtr);

    Var     aktResFile  : Integer;

    Procedure Init_ScrollBar(Dialog: DialogPtr);

        Var     Box     : Rect;
                Scroll  : ControlHandle;
        Begin
          DItem_GetBox(Dialog,MacroScrollBar,Box);
          Scroll := NewControl(Dialog,Box,'',true,0,0,1,16,0);
          SetCRefCon(Scroll,Ord4(Dialog));
          Globals^^.TextScroll := Scroll;
        end;

    Function neue_Datei(DateiName: Str255): Integer;

        Var     H       : Handle;
                Demo    : Str255;
                RefNr   : Integer;
                Err     : OSErr;
        Begin
          CreateResFile(DateiName); {Resource-File neu anlegen}
          RefNr := OpenResFile(DataName);
          UseResFile(RefNr);
          Demo := 'Demo-Text';
          Err := PtrToHand(@Demo,H,Succ(Length(Demo))); {Demo --> H^^}
            {Handle in Resource-File eintragen:}
          AddResource(H,'STR ',1,'Demo-Name');
          neue_Datei := RefNr;
        end;

    Procedure MacroListe_laden;

        Var     Index       : Integer;
                Dimension   : Rect; {Liste}
                Zelle       : Point;
                MacroName   : Str255;
                Abmessungen : Rect;

        Procedure Get_ResName(var Name: Str255; Index: Integer);

            Var     ResHandle   : Handle;
                    ResId       : Integer;
                    theType     : ResType;
            Begin
              ResHandle := GetIndResource('STR ',Index);
              GetResInfo(ResHandle,ResId,theType,Name);
            end;

        Begin {MacroListe_laden}
          DItem_GetBox(Globals^^.Dialog,ListenBox,Abmessungen);
          UseResFile(Globals^^.DataRefNr);
          Index := Count1Resources('STR ');
          SetRect(Dimension,0,0,1,Index); {Dimension der Liste}
          Zelle.v := 0; Zelle.h := 0; {links oben}
          with Globals^^ do {neue Liste anlegen}
        Liste := LNew(Abmessungen,Dimension,Zelle,0,Dialog,true,false,false,true);
          Zelle.h := 0; {1. (einzige) Spalte}
          while Index > 0 do
            Begin
              Get_ResName(MacroName,Index);
              Index := Pred(Index);
              Zelle.v := Index;
                {Macro in Liste eintragen:}
              LSetCell(@MacroName[1],Length(MacroName),Zelle,Globals^^.Liste);
            end;
        end;{MacroListe_laden}


    Begin {Init_Items}
      DItem_SetText(Dialog,MacroTextBox,'(Dieser Text wird gesendet)');
      DItem_Hilite(Dialog,SendeKnopf,false);
      DItem_Hilite(Dialog,LoeschKnopf,false);
      DItem_Hilite(Dialog,KorrekturKnopf,false);
      HideDItem(Dialog,MacroIdBox);
      HideDItem(Dialog,okKnopf);
      HideDItem(Dialog,ZurueckKnopf);
      HideDItem(Dialog,AbbruchKnopf);
      Globals^^.PortRefNr := AoutRefNum; {Default = Modemanschluß}
      DItem_SetCtlValue(Dialog,ModemKnopf,1); {Modemknopf einschalten}
      Init_ScrollBar(Dialog);
      aktResFile := CurResFile;
      with Globals^^ do
```

```
      Begin
        DataRefNr := OpenResFile(DataName); {DFÜ-Resourcedatei öffnen}
        If DataRefNr < 0 then {Datei nicht vorhanden --> neu anlegen}
          DataRefNr := neue_Datei(DataName);
      end;
    MacroListe_laden;
    UseResFile(aktResFile);
  end; {Init_Items}

Begin {DrvrOpen}
  If SysCom^.DctlWindow = NIL then { DFÜ-Fenster wird erstmalig geöffnet }
    Begin
      Dialogfenster_oeffnen;
      HLock(SysCom^.DctlStorage);
      Globals := Pointer(SysCom^.DctlStorage);
      with Globals^^ do {globale Variablen initialisieren}
        Begin
          Dialog := Pointer(SysCom^.DctlWindow);
          TextResHdl := NIL;
          SendeHdl := NewHandle(0);
          Init_Items(Dialog);
          ShowWindow(Dialog);
        end;
      HUnlock(SysCom^.DctlStorage);
    end;
  DrvrOpen := 0;
end; {DrvrOpen}
{_______________________________________________________________________}
Function DrvrControl(SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;

  Var    Trick   : Record case integer of
                     0: (CSParam: array[0..1] of Integer);
                     1: (EventPtr: ^EventRecord)
                   end;
         Globals : Global_Hdl;

Procedure EventCall(Event: EventRecord);

  Var    Dialog  : DialogPtr;
         ItemNr  : Integer;

  Procedure update_DA;

    Var    LRect   : Rect;

    Begin
      LRect := Globals^^.Liste^^.RView;
      InsetRect(LRect,-1,-1); { Box um ein Pixel vergrößern}
      FrameRect(LRect);
      LUpdate(Globals^^.Dialog^.visRgn,Globals^^.Liste);
    end; { of UpdateDA }

  Procedure activate_DA(SysCom: DCtlPtr);

    var    MenuH   : menuHandle;

    begin
      MenuH := GetMenu(SysCom^.DCtlMenu);
      MenuH^^.menuID := SysCom^.DCtlMenu;
      InsertMenu(MenuH,0); {DFÜ-Menü an aktuelle Menüleiste anhängen}
      DrawMenuBar;
      LActivate(true,Globals^^.Liste);
    end; { of ActivateDA }

Procedure senden;

  Var    Txt : Str255;
         Err : OSErr;
  Begin
    with Globals^^ do
      Begin
        DItem_GetText(Dialog,MacroTextBox,Txt);
        Err := PtrAndHand(@Txt[1],SendeHdl,Length(Txt));
        HideDItem(Dialog,SendeKnopf);
        ShowDItem(Dialog,AbbruchKnopf);
      end;
  end;

Procedure loeschen;

  Var    Zelle       : Point;
         aktResFile  : Integer;
         ResHdl      : Handle;
  Begin
    Zelle.v := 0; Zelle.h := 0;
    If LGetSelect(true,Zelle,Globals^^.Liste) then
      Begin
        aktResFile := CurResFile;
        UseResFile(Globals^^.DataRefNr);
        ResHdl := Get1IndResource('STR ',Succ(Zelle.v));
        RmveResource(ResHdl);      { Resource entfernen }
        ChangedResource(ResHdl);   { Resourcefile als geändert kennzeichnen}
        writeResource(ResHdl);     { Resourcefile aktualisieren }
        UseResFile(aktResFile);
        LDelRow(1,Zelle.v,Globals^^.Liste); { Listeneintrag entfernen }
      end;
  end;

Procedure update_EditFeld;

  Var    aktResFile  : Integer;
         SendeText   : Str255;
         Zelle       : Point;

  Procedure aktuellen_Text_sichern;

   Var    Txt     : Str255;

   Begin
     with Globals^^ do
       if TextResHdl <> NIL then
        Begin
          DItem_GetText(Dialog,MacroTextBox,Txt);
          LoadResource(TextResHdl);
          SetHandleSize(TextResHdl,Succ(Length(Txt)));
          HLock(TextResHdl);
          BlockMove(@Txt,TextResHdl^,Succ(Length(Txt)));
          HUnlock(TextResHdl);
          ChangedResource(TextResHdl);
        end;
   end;

 Begin {update_EditFeld}
   aktuellen_Text_sichern;
   Globals^^.TextResHdl := NIL;
   Zelle.v := 0; Zelle.h := 0;
   If not LGetSelect(true,Zelle,Globals^^.Liste) then SendeText := ''
     else
```

```
    Begin   { aktiviertes Macro in der Textbox anzeigen }
      aktResFile := CurResFile;
      UseResFile(Globals^^.DataRefNr);
      Globals^^.TextResHdl := Get1IndResource('STR ',Succ(Zelle.v));
      HLock(Globals^^.TextResHdl);
      BlockMove(Globals^^.TextResHdl^,@SendeText,SizeOf(SendeText));
      HUnlock(Globals^^.TextResHdl);
      UseResFile(aktResFile);
    end;
  DItem_SetText(Globals^^.Dialog,MacroTextBox,SendeText);
 SelIText(Globals^^.Dialog,MacroTextBox,Length(SendeText),Length(SendeText));
  DrawDialog(Globals^^.Dialog);
 end; {update_EditFeld}

Procedure Change_Items(Dialog: DialogPtr);
                         { Bedienungsknöpfe deaktivieren, }
  Var    Txt : Str255;   { Editierknöpfe aktivieren }

  Begin
    DItem_Hilite(Dialog,SendeKnopf,false);
    DItem_Hilite(Dialog,LoeschKnopf,false);
    DItem_Hilite(Dialog,KorrekturKnopf,false);
    DItem_Hilite(Dialog,NeuKnopf,false);
    HideDItem(Dialog,MacroTextBox);
    HideControl(Globals^^.TextScroll);
    HideDItem(Dialog,MacroScrollBar);
    HideDItem(Dialog,ListenBox);
    HideDItem(Dialog,sendenText);
    ShowDItem(Dialog,MacroIdBox);
    ShowDItem(Dialog,okKnopf);
    ShowDItem(Dialog,ZurueckKnopf);
    DItem_GetText(Dialog,MacroIdBox,Txt);
    SelIText(Dialog,MacroIdBox,0,Length(Txt));
    DrawDialog(Dialog);
  end; {Change_Items}


Procedure neu(Dialog: DialogPtr);
                              { neuen Eintrag vorbereiten }
  Var    Zelle   : Point;
         Select  : Boolean;
  Begin
    repeat {alle eventuell aktivierten Listeneinträge deaktivieren}
      Zelle.v := 0; Zelle.h := 0;
      Select := LGetSelect(true,Zelle,Globals^^.Liste);
      If Select then LSetSelect(false,Zelle,Globals^^.Liste);
    until not Select;
    update_EditFeld;       { alten Text sichern }
    Change_Items(Dialog); { Editierknöpfe aktivieren }
  end;

Procedure Liste(Event: EventRecord; Dialog: DialogPtr);
                                  { Mausklick im Listenfenster }
  Var    Zelle           : Point;
         MausPos         : Point;    { Position des Mausklicks }
         DoppelKlick     : Boolean;
         Treffer         : Boolean;  { wurde Macro angeklickt? }
  Begin
    MausPos := Event.where;
    GlobalToLocal(MausPos);
    DoppelKlick := LClick(MausPos,Event.modifiers,Globals^^.Liste);
    Zelle.v := 0; Zelle.h := 0;
    Treffer := LGetSelect(true,Zelle,Globals^^.Liste); {auf Macro?}
    DItem_Hilite(Dialog,SendeKnopf,Treffer); {ggf. Knöpfe aktivieren}
    DItem_Hilite(Dialog,LoeschKnopf,Treffer);
    If DoppelKlick then senden;
  end;

Procedure Zurueck;       { Editierknöpfe deaktivieren, }
                         { Bedienungsknöpfe aktivieren }
  Begin
    DItem_Hilite(Dialog,NeuKnopf,true);
    HideDItem(Dialog,MacroIdBox);
    HideDItem(Dialog,okKnopf);
    HideDItem(Dialog,ZurueckKnopf);
    ShowDItem(Dialog,SendeKnopf);
    ShowDItem(Dialog,ListenBox);
    ShowDItem(Dialog,MacroScrollBar);
    ShowControl(Globals^^.TextScroll);
    ShowDItem(Dialog,sendenText);
    ShowDItem(Dialog,MacroTextBox);
    DrawDialog(Dialog);
  end;

Procedure Editiermodus_abschliessen;     { Texteingabe abschließen und }
                                         { eingegebenen Text sichern }
  Var    aktResFile      : Integer;
         neues_Macro     : Str255;
         Zelle           : Point;

  Procedure eintragen(MacroName: Str255); {speichern}

    Var    aktResFile  : Integer;
           Zelle       : Point;
    Begin
      Globals^^.TextResHdl := NewHandle(1); {Leerstring}
        { neue Macrobezeichnung in Resourcefile eintragen:}
      aktResFile := CurResFile;
      UseResFile(Globals^^.DataRefNr);
      AddResource(Globals^^.TextResHdl,'STR ',UniqueID('STR '),MacroName);
      Zelle.h := 0;
      Zelle.v := Count1Resources('STR ');
      UseResFile(aktResFile);
        { Liste ergänzen: }
      Zelle.v := LAddRow(1,Zelle.v,Globals^^.Liste);
      LSetCell(@MacroName[1],Length(MacroName),Zelle,Globals^^.Liste);
      LSetSelect(true,Zelle,Globals^^.Liste); {Macro aktivieren}
      LAutoScroll(Globals^^.Liste);  {ggf. scrollen, damit sichtbar}
    end;

  Procedure korrigieren(neues_Macro: Str255; Zelle: Point);

    Var    ID_Nr          : Integer;
           Typ            : ResType;
           altes_Macro    : Str255;
    Begin
      GetResInfo(Globals^^.TextResHdl,ID_Nr,Typ,altes_Macro);
      SetResInfo(Globals^^.TextResHdl,ID_Nr,neues_Macro);
      ChangedResource(Globals^^.TextResHdl); {als geändert kennzeichnen}
      LSetCell(@neues_Macro[1],Length(neues_Macro),Zelle,Globals^^.Liste);
    end;

  Begin {Editiermodus_abschliessen}
    DItem_GetText(Globals^^.Dialog,MacroIdBox,neues_Macro);
    Zelle.v := 0; Zelle.h := 0;
    If LGetSelect(true,Zelle,Globals^^.Liste) {Eintrag existiert bereits}
          then korrigieren(neues_Macro,Zelle)
          else eintragen(neues_Macro);
    Zurueck; {Editknöpfe aus, Bedienknöpfe ein}
  end;
```

```
Procedure Hilfe;

    Var     Dialog1,
            Dialog2 : DialogPtr;
            BaseId  : Integer; {ID-Nummer der ersten DLOG-Resource}
            Item    : Integer;
    Begin
      BaseID := $BFE0 - 32*(SysCom^.DCtlRefNum);{ vom Font/DA Mover geändert!}
      Dialog1 := GetNewDialog(BaseId+1,NIL,Pointer(-1));
      ModalDialog(NIL,Item);
      Dialog2 := GetNewDialog(BaseId+2,NIL,Pointer(-1));
      DisposDialog(Dialog1);     { immer erst neues Fenster aufbauen, }
      ModalDialog(NIL,Item);     { dann altes Fenster löschen! }
      Dialog1 := GetNewDialog(BaseId+3,NIL,Pointer(-1));
      DisposDialog(Dialog2);
      ModalDialog(NIL,Item);
      DisPosDialog(Dialog1);
    end;

Procedure einschalten(ItemNr: Integer); { umschalten zwischen }
                                        { Modem-/Druckerknopf }
    Var     N       : Integer;

    Begin
      For N := ModemKnopf to DruckerKnopf do {erst mal alles ausschalten}
            DItem_SetCtlValue(Globals^^.Dialog,N,0);
      DItem_SetCtlValue(Globals^^.Dialog,ItemNr,1);
      with Globals^^ do case ItemNr of
                          ModemKnopf   : PortRefNr := AoutRefNum;
                          DruckerKnopf : PortRefNr := BoutRefNum;
                        end;
    end;

Procedure Korrektur(Dialog: DialogPtr); {umschalten auf Editiermodus}

    Var     Id      : Integer;
            Typ     : ResType;
            Macro   : Str255;
    Begin
      GetResInfo(Globals^^.TextResHdl,Id,Typ,Macro);
      DItem_SetText(Dialog,MacroIdBox,Macro);
      Change_Items(Dialog);
    end;

Procedure Scroll_EditText(where: Point);

    Procedure ThumbControl(Ctrl: ControlHandle; where: Point);

        Var     alter_Wert  : Integer;
                Differenz   : Integer;
                DiaPeek     : DialogPeek;
        Begin
          alter_Wert := GetCtlValue(Ctrl);
          If TrackControl(Ctrl,where,NIL) = inThumb then
            Begin
              Differenz := alter_Wert - GetCtlValue(Ctrl);
              DiaPeek := Pointer(GetCRefCon(Ctrl));
              TEScroll(0,Differenz,DiaPeek^.TextH);
            end;
        end;

    Begin
      GlobalToLocal(where);
      If TestControl(Globals^^.TextScroll,where) = inThumb
        then ThumbControl(Globals^^.TextScroll,where)
        else if TrackControl(Globals^^.TextScroll,where,@V_Action) > 0 then;
    end;

Procedure Check_Liste;

    Var     Zelle       : Point;
            aktiv       : Boolean;      { Macro aktiviert? }
            DiaPeek     : DialogPeek;
            TxtHdl      : TEHandle;
            Offset,
            GesamtHoehe : Integer; { von: destRect der SendeDaten}
            F_Info      : FontInfo;
            Scroll      : ControlHandle;
            Zeilen      : Integer;
            SendeDaten  : Str255;
    Begin
      Zelle.v := 0; Zelle.h := 0;
      with Globals^^ do
        Begin
          aktiv := LGetSelect(true,Zelle,Liste);
          DItem_Hilite(Dialog,SendeKnopf,aktiv);
          DItem_Hilite(Dialog,KorrekturKnopf,aktiv);
          DItem_Hilite(Dialog,LoeschKnopf,aktiv);
          DItem_GetText(Dialog,MacroTextBox,SendeDaten);
          DiaPeek := Pointer(Dialog);
        end;
          { Wieviel Platz benötigen die Sendedaten?}
      GetFontInfo(F_Info);
      TxtHdl := DialogPeek(Dialog)^.TextH;
      with TxtHdl^^ do
        Begin
          Offset := viewRect.top - destRect.top;
          Zeilen := NLines - 3;
        end;
      If SendeDaten[Length(SendeDaten)] = chr(13) {Return am Ende} then
                Zeilen := Succ(Zeilen); {Cursor auf nächste Zeile}
      GesamtHoehe := Zeilen*(F_Info.Descent+F_Info.Ascent+1);
      Scroll := Globals^^.TextScroll;
      If (Offset = 0) and {???}
         (GesamtHoehe <= 0) then HiliteControl(Scroll,255) {ausschalten}
                            else
               Begin
                 If GesamtHoehe < Offset then GesamtHoehe := Offset;
                 SetCtlMax(Scroll,GesamtHoehe);
                 SetCtlValue(Scroll,Offset); {nur wegen Edit-Aktionen}
                 HiliteControl(Scroll,0);
               end;
    end;

Function KeyPress(Key: Integer): Boolean;

    Begin
      KeyPress := BitAnd(Event.modifiers,Key) = Key;
    end;

Begin
  If (Event.what = KeyDown) and
     KeyPress(CmdKey) then SysBeep(1) {keine Kurzbefehle möglich!}
else if DialogSelect(Event,Dialog,ItemNr) then
    Case ItemNr of
      SendeKnopf    : senden;
      LoeschKnopf   : If Keypress(CmdKey) then loeschen
                                          else SysBeep(1);
      NeuKnopf      : neu(Dialog);
      MacroTextBox  : ; {wird von DIALOGSELECT übernommen}
      ListenBox     : Begin
                        Liste(Event,Dialog);
                        update_EditFeld;
                      end;
```

```
        ScrollBox       : Liste(Event,Dialog);
        MacroIdBox      : If Event.what = KeyDown {Tastendruck} then
                            if Event.Message Mod 256 = 13 {Returntaste} then
                              Editiermodus_abschliessen;
        okKnopf         : Editiermodus_abschliessen;
        ZurueckKnopf    : Zurueck;
       HilfeKnopf       : Hilfe;
       ModemKnopf,
       DruckerKnopf     : einschalten(ItemNr);
       KorrekturKnopf   : Korrektur(Dialog);
       MacroScrollBar   : Scroll_EditText(Event.where);
       AbbruchKnopf     : SetHandleSize(Globals^^.SendeHdl,0);
      end
    else case Event.what of
         updateEvt    : Update_DA;
         activateEvt  : if odd(Event.modifiers) then Activate_DA(SysCom)
                                                 else Deactivate_DA(SysCom);
     end;
    Check_Liste;
  end;

Procedure Info;     {für Menüpunkt 'Info...'}

  Var     Event   : EventRecord;
          Window  : WindowPtr;
          R       : Rect;
  Begin
    SetRect(R,100,100,300,300);
    Window := NewWindow(NIL,R,'',true,1,Pointer(-1),false,0);
    SetPort(Window);
    MoveTo(50,50);
    DrawString('DFÜ-Hilfe 2.1');
    MoveTo(50,100);
    DrawString('A.Giesel 1987/88');
    repeat
      If GetNextEvent(EveryEvent,Event) then;
    until Event.what = MouseDown;
    DisposeWindow(Window);
  end;

Procedure Hdl_Idle;

  Begin
    with DialogPeek(Globals^^.Dialog)^ do
      If editField >= 0 {Text existiert} then
           TEIdle(TEHandle(TextH)); { Cursor blinken lassen }
  end;

Procedure Pruef_Sendung;

  Type    Memory   = Packed Array [0..255] of Byte;

  Var     Rest     : Integer; {Anzahl noch zu sendender Zeichen}
          MemPtr   : ^Memory; {Maske}
          CR       : Boolean;
          Err      : OSErr;
          N        : LongInt;
  Begin
    with Globals^^ do
      Begin
        Rest := GetHandleSize(SendeHdl);
        If Rest > 0 then {noch Daten zu senden}
          Begin
            HLock(SendeHdl);
            MemPtr := Pointer(SendeHdl^);
            N := 0;
            repeat {Zeilenende suchen}
              CR := MemPtr^[N] = 13;
              N := N + 1;
            until (N >= Rest) or CR;
            Err := FSWrite(PortRefNr,N,SendeHdl^); {Zeile senden}
            Rest := Rest - N;
            If Rest > 0 then {Rest nach vorn holen}
              BlockMove(@MemPtr^[N],SendeHdl^,Rest);
            HUnlock(SendeHdl);
            SetHandleSize(SendeHdl,Rest);
          end;
        If Rest = 0 then
          Begin
            HideDItem(Dialog,Abbruchknopf);
            ShowDItem(Dialog,SendeKnopf);
          end;
      end;
  end;

  Begin {DrvrControl}
    DrvrControl := 0; {Default}
    HLock(SysCom^.DCtlStorage);
    Globals := Pointer(SysCom^.DctlStorage);
    SetPort(Globals^^.Dialog);
    case SystemParameter^.csCode of
       accEvent    : Begin { Event bearbeiten }
                       Trick.CSParam[0] := SystemParameter^.CSParam[0];
                       Trick.CSParam[1] := SystemParameter^.CSParam[1];
                       EventCall(Trick.EventPtr^);
                     end;
       accRun      : Pruef_Sendung;
       accCursor   : Hdl_Idle;
       accMenu     : Begin
                       case SystemParameter^.CSParam[1] of
                         1: Info; { menu item #1 }
                       end;
                       HiliteMenu(0);
                     end;
       accUndo:    ;
       accCut:     DlgCut(Globals^^.Dialog);
       accCopy:    DlgCopy(Globals^^.Dialog);
       accPaste:   DlgPaste(Globals^^.Dialog);
       accClear:   DlgDelete(Globals^^.Dialog);
     end;
     HUnLock(SysCom^.DCtlStorage);
  end; {DrvrControl}
{_______________________________________________________________________}
Function DrvrClose(SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;

  Var     Globals : Global_Hdl;

  Begin {alles schließen bzw. freigeben}
    Globals := Pointer(SysCom^.DctlStorage);
    CloseResFile(Globals^^.DataRefNr);
    Deactivate_DA(SysCom);
    LDispose(Globals^^.Liste);
    DisposDialog(Globals^^.Dialog);
    SysCom^.DctlWindow := NIL;
    DisposHandle(Handle(SysCom^.DCtlStorage));
  END; (* of Close *)

Function DrvrStatus (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
  Begin DrvrStatus := statusErr; end;
Function DrvrPrime  (SystemParameter:  ParmBlkPtr; SysCom: DCtlPtr): OSErr;
  Begin DrvrPrime := unimpErr; end;

end.
```

Hilfe von UseResFile zuvor auch sicherstellen. Sicherheitshalber sollte aber zuerst die ID-Nummer der gerade aktiven Resourcedatei mit CurResFile gesichert und diese nach Initialisierung der Liste wieder aktiviert werden! Ist die Liste definiert, braucht man nur noch mittels einer einfachen Schleife (und mit Hilfe von Get1IndResource und GetResInfo) die Namen der Macro-Resourcen – die ja identisch mit den vom Benutzer vergebenen Kennwörtern sind – in die Liste einzutragen.

Aufbau und Funktion der DFÜ-Hilfe

Das „Hauptprogramm" der DFÜ-Hilfe, die DrvrControl-Funktion, besteht im wesenllichen aus einer CASE-Konstruktion, in der die für den jeweiligen Aufruf zuständigen Programmteile gerufen werden. Um den für die DA-Prozedur EventCall benötigten Zeiger auf den zu bearbeitenden Event zu erhalten, wird übrigens ein alter Pascal-Trick benützt: Da der Compiler den verschiedenen CASE-Zweigen eines varianten Records den selben Speicherplatz zuweist, ist auf diese Weise eine einfache Typumwandlung möglich (siehe Variable Trick). Das Fenster der DFÜ-Hilfe ist als Dialogfenster (DLOG) definiert. Dadurch kann man sich durch Einsatz der Dialogmanager-Prozedur DialogSelect die Programmierung wesentlich vereinfachen: Diese Prozedur übernimmt nämlich nicht nur die Bearbeitung sämtlicher Events, sondern liefert ggf. auch noch die Nummer eines angeklickten Dialogelementes! Lediglich updateEvents und activateEvents bedürfen noch einer kleinen Nachbehandlung durch Listmanager-Prozeduren. *Bild 3* zeigt die DFÜ-Hilfe in Aktion. Ausschnitt ① enthält die vom List-Manager verwalteten Kennwörter (= Namen der Macroresourcen). Rechts daneben befindet sich die zugehörige Scrollbar, die vom List-Manager automatisch aktiviert und verwaltet wird, falls sich mehr Einträge in der Liste befinden als in den Ausschnitt passen. Im Ausschnitt ② darunter befindet sich das im Ausschnitt ① aktivierte Macro, wo es jederzeit korrigiert werden kann. Auch hier befindet sich rechts daneben eine Scrollbar, die allerdings nicht so komforabel von einem Manager verwaltet wird. Die hierfür zuständige Prozedur Scroll_EditText stellt bei einem MouseDown-Event zunächst fest, ob sich die Maus im „Aufzug" befindet (in Thumb). Ist dies der Fall, kommt ThumbControl zum Zuge, andernfalls TrackControl. Letztere bekommt als Parameter die Adresse der Prozedur V_Action übergeben, welche die CtlValue der Scrollbar entsprechend der gewünschten Aktion aktualisiert. Rechts oben kann zwischen den Ports umgeschaltet werden, die Funktion der Knöpfe darunter kann den Resourcedefinitionen entnommen werden (Resource 'DITL' (BasisId+2)). Diese Funktionsbeschreibung und weitere Hinweise erscheinen bei aktivierter DFÜ-Hilfe nach Anklicken des Hilfe-Knopfes in einem separaten Fenster. Eine besondere Funktion hat der Knopf NEU. Nach dem Anklicken verändert sich das Erscheinungsbild der DFÜ-Hilfe etwas (Abb.4). Unterhalb der Macroliste kann jetzt ein neues Kennwort eingegeben werden. Nach Drücken der Returntaste oder Anklicken des OK-Knopfes wird es in die Kennwortliste übernommen, und das Erscheinungsbild wechselt wieder in die gewohnte Form. Anschließend kann im Ausschnitt ② das eigentliche Macro eingetragen werden. Besondere Beachtung verdient noch die Prozedur Pruef_Sendung. Sie wird im Sekundenabstand gerufen und übernimmt die Ausgabe der Macrodaten über die serielle Schnittstelle. Damit das empfangende Modem genug Zeit zur Auswertung der empfangenen Sequenz hat, wird bei mehrzeiligen Macros immer nur eine Zeile pro Aufruf übertragen. Während der Übertragung des Macros wird der „Senden"-Knopf unsichtbar gemacht und der darunter liegende „Abbruch"-Knopf aktiviert. Mit Ihm kann die Übertragung jederzeit gestoppt werden.

Übersetzen und Linken

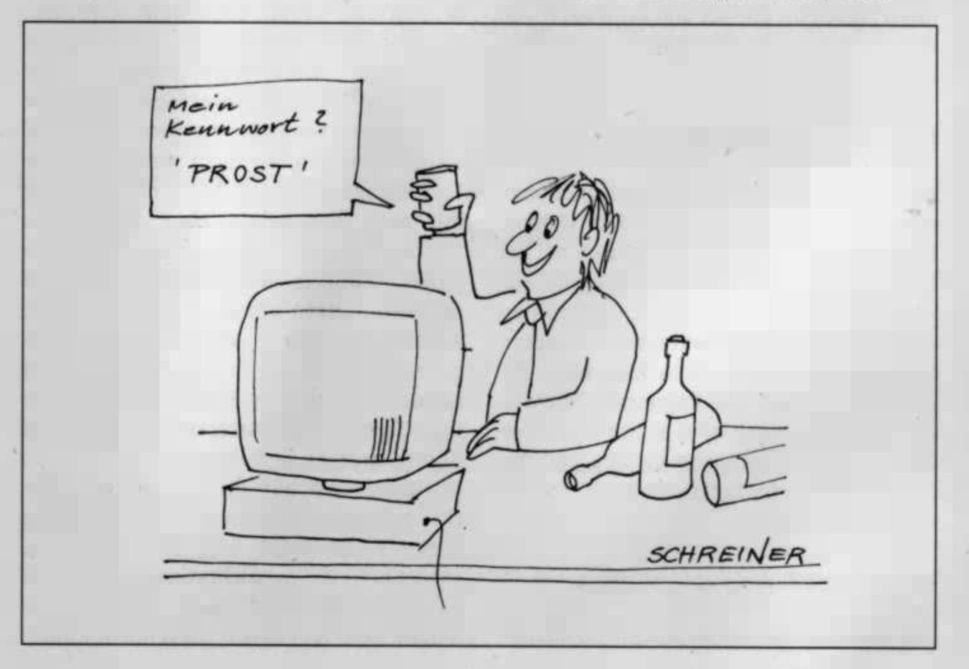
Das Make-File in Bild 5 übernimmt im MPW den kompletten Übersetzungs- und Linkvorgang, und wird am Besten über das Build-Menü gestartet. Voraussetzung ist natürlich, daß der Pascal-Quelltext unter „DFÜ-Hilfe.P" und der Resource-Quelltext unter „DFÜ-Hilfe.R" gespeichert ist. Als besonderer Service wird einem unter MPW nach erfolgreichem Abschluß der ganzen Aktion die komplette Befehlssequenz zum Starten des Font/DA Movers angeboten, und nach dem Drücken der Enter-Taste braucht man nur noch die (automatisch geladene) DFÜ-Hilfe zu selektieren und auf 'Kopieren' zu klicken.

Für TML (und andere Compiler) sind einige Änderungen in den Quelltexten nötig. Das Wort 'Unit' ist durch 'Program' zu ersetzen, alles zwischen 'Interface' und 'Implementation' (einschließlich) ist zu entfernen. Alle mit 'Drvr' beginnenden Funktionen sind in Prozeduren umzuwandeln, und das 'Drvr' ist aus ihrem Namen zu entfernen. Anschließend muß noch die (jetzige) Prozedur 'Control' in 'Ctl' umbenannt werden.

Bei den Resourcedefinitionen sind umfangreichere Formatumwandlungen nötig, die aber nur Arbeit und keine Probleme machen dürften. Am Besten orientiert man sich dabei an den TML-Beispielen. Auf eine Besonderheit soll hier noch hingewiesen werden: MPW betrachtet bei allen Zahlenwerten die Kombination '0x' als Beginn einer Hexzahl und '0b' als Beginn einer Binärzahl. Übersetzt wird in TML mit der Option 'Create Desk Accessory'. Installiert wird das DA auch hier mit dem Font/DA-Mover, der hier allerdings „von Hand" gestartet werden muß. Mit Hilfe des rechten 'öffnen'-Knopfes öffnet man die DFÜ-Hilfe, selektiert sie im rechten Ausschnitt und klickt anschließend auf 'kopieren' – fertig!

Literatur

Schirmacher, Arne: Das große Macintosh-Arbeitsbuch, Franzis-Verlag, München, 1987.



Albert Meier

Ein Betriebssystem in Modula-2

Teil 1: Ein/Ausgabe-Fenster

Das Interesse an dem Artikel „Parallele Prozesse in Modula-2“ (mc, Ausgabe 6/88) war groß. Viele Leser wollen die Anwendungen dieser Techniken in größerem Rahmen studieren: ganze Projekte sind gefragt. In dieser Serie beschreiben wir ein Betriebssystem, das in Modula-2 geschrieben ist.

Damit dieser Aufsatz nicht nur für diejenigen Leser interessant ist, die selbst ein Betriebssystem schreiben wollen, sondern besonders auch für all jene, die ein solches benutzen und ein wenig hinter die Kulissen blicken möchten, haben wir eine unter Programmierern beliebte Form der Beschreibung gewählt. Wir gehen von einem Programmbeispiel aus, welches verschiedene Funktionen des zugrundeliegenden Betriebssystems nutzt, und erläutern die interessanten Funktionen am Beispiel. Wir haben uns erlaubt, viele Details wegzulassen, die bestimmt für den einen oder anderen sehr interessant sind, dies zugunsten von Gedanken, die in der heutigen Zeit noch nicht sehr geläufig sind. Insbesondere ist es für jedermann möglich, die Gedanken aus dieser Serie aufzunehmen und eigene Ideen einzubringen. Schließlich handelt es sich bei den nun folgenden Ausführungen nicht nur um ein trockenes Gedankenwerk, sondern um ein lebendiges Beispiel, welches der Leser selber nachvollziehen kann. Das beschriebene Betriebssystem ist seit mehreren Jahren im Einsatz und kann sogar von interessierten PC-Benutzern erworben werden (siehe Kasten).

Die Schnittstelle muß passen

Ein Betriebssystem ist das Bindeglied zwischen einem Programm und dem Computer. Man kann es mit den Schienen bei der Eisenbahn vergleichen. Der fahrende Zug ist das Programm, und der Schienenuntergrund, das ist die Hardware, der Computer. Ohne Schienen fährt kein Zug, ohne Betriebssystem läuft kein Programm. Wir wollen zu Beginn noch eine Weile beim Vergleich mit der Eisenbahn bleiben und uns Kriterien für ein funktionierendes Schienennetz ausdenken.

Das Fahrgestell des Zuges ist genau auf die Schienenweite eingestellt, sonst paßt der Zug nämlich gar nicht auf die Schienen. Ebenfalls sind Schienen nirgends unterbrochen, da sonst der Zug herausfällt. Das ist logisch und reicht aus, damit der Zug fahren kann. Wir wollen jedoch noch mehr: der Zug soll sinnvoll, steuerbar fahren; wir wollen mit der Eisenbahn mal hierhin, mal dorthin reisen und Zugkompositionen sollen beliebig zusammenstellbar sein. Also benötigen wir Steuerelemente, Weichen und Verbindungen, Bahnhöfe und Bahnsteige, die es dem Reisenden erlauben, mit dem Zug optimal sein Ziel zu erreichen. Dazu soll unser Schienennetz natürlich auch flexibel sein, da wir ja nicht nur Reisende, sondern auch Güter auf der gleichen Schnittstelle, den Gleisanlagen, befördern wollen. Wir wollen das Schienennetz für unterschiedliche Aufgaben einsetzen, Schnellzüge fahren über weite Strecken, Regionalzüge bedienen alle Stationen, in Güterzügen wird Ware befördert und manchmal will man sogar eine Extrafahrt mit einem Dampfzug machen. Selbstverständlich wollen wir nicht warten, bis der Schnellzug am Ziel angekommen ist, bis wir mit dem Regionalzug über die gleiche Strecke losfahren können, der Schienenstrang sollte möglichst gleichzeitig von mehreren Zügen genutzt werden können. Allerdings verlangen wir, daß Unfälle weitgehend ausgeschlossen sein müssen. Dazu werden noch weitere Einrichtungen, benötigt: Signale, Blockstrecken, Steuer- und Kontrollanlagen und Regeln.

Wir haben uns nun die wichtigsten Kriterien ausgemalt und bleiben noch für eine kurze Weile beim Bild der Eisenbahn, schließlich ist es nicht unwichtig, auch Aussagen über die Qualität des Schienennetzes und die daraus folgenden Konsequenzen machen zu können. Wir haben bisher bloß von der Schnittstelle, den Gleisanlagen, nicht jedoch von den Zügen selbst gesprochen. Doch ist offensichtlich, daß die Qualität der Eisenbahn in hohem Maß von der Qualität der Gleise abhängt. Das Rollmaterial kann noch so gut und modern gebaut sein, wenn das Gleis, die Schnittstelle zum Boden, ungenügend ist, so hat dies Komforteinbußen, wenn nicht gar Sicherheitseinbußen zur Folge. Von der Qualität der Gleisanlagen hängen der Komfort und die Sicherheit beim Eisenbahnfahren in erster Linie ab. Natürlich können diese Qualitäten durch mangelhafte oder veraltete Züge wieder zunichte gemacht werden.

Die Qualität einer Schnittstelle

Analog zur Eisenbahn können wir nun die obigen Gedanken auch für Betriebssysteme anwenden. Auch hier gilt: je besser das Betriebssystem, desto komfortabler und sicherer kann das Programm darauf laufen. Als Schnittstelle zum Computer muß demnach das Betriebssystem, analog zum Schienennetz, den darauf laufenden Programmen den nötigen Halt geben, es muß zum Beispiel genau definiert sein und darf keine Löcher aufweisen (z. B. Situationen, die nicht definiert sind), damit die Programme nicht abstürzen. Analog zu den Weichen und Steuerelementen stellt ein Betriebssystem die Leistung des Computers dem Programm in steuerbarer Form zur Verfügung, z. B. als aufrufbare Funktionen oder als Informationen, die den weiteren Verlauf beeinflussen können.

Selbstverständlich muß das Betriebssystem es zulassen, daß Programme für ganz unterschiedliche Zwecke darauf laufen. Und nicht zuletzt ist es sinnvoll, wenn das Betriebssystem Mechanismen vorsieht, die es erlauben, mehrere Programme gleichzeitig zu betreiben. Bei all dem wird natürlich verlangt, daß es weitgehend ausgeschlossen ist, daß das Betriebssystem mit einer Fehlermeldung abbricht.

Bevor wir auf einzelne Punkte des Betriebssystems eingehen, sei Ihnen ein Überblick über den Funktionsumfang des beschriebenen Systems geben.

Überblick

Die Farsight Modula Toolbox offeriert Funktionen für Ein-/Ausgabe über Bildschirm und Drucker, Menü- und Fensterverwaltung, Dateiverwaltung, die Integration neuer Programme, die Datenübertragung zwischen Programmen, virtuelle Speicherverwaltung, Makroerzeugung und -verwaltung, Hintergrundprozesse, Farbeinstellung, Datum und Zeit, Hilfetexte, Fehlermeldungen, Ton, Grafik, usw. In unserem Beispiel wollen wir vor allem die Funktionen für die Ein-/Ausgabe, die parallelen Prozesse und die virtuelle Speicherverwaltung kennenlernen. Anhand der Erläuterungen möchten wir aufzeigen, daß es mit einem komfortablen Betriebssystem möglich ist, mit wenig Programmieraufwand ansprechende und leistungsfähige Programme mit einer einheitlichen Benutzeroberfläche und vollkommener Integration verschiedener unterschiedlicher Programme zu schreiben. Gleichzeitig möchten wir begründen, weshalb Modula-2 für die Entwicklung gewählt wurde: Es ist an dem Beispiel mit der Eisenbahn ersichtlich, daß vor allem die exakte Definition der Schnittstelle des Betriebssystems für den eigentlichen Betrieb wichtig ist. Dank des Modulkonzeptes von Modula-2 ist mit der Festlegung der Definitionsmoduln diese Forderung nämlich schon weitgehend erfüllt.

Ein-/Ausgabe im Fenster

Unserem Beispielprogramm SAM (für Sample = Beispiel) entnehmen wir einen Teil des Moduls SAMdialogue, um gleichzeitig einige Ein-/Ausgabefunktionen und deren Wirkung vorzuführen.

Bild 1 zeigt Definitions- und Implementationsmodul von SAMdialogue, *Bild 2* die verwendete Dialogdefinition und *Bild 3* das Programm bei der Ausführung. Wir entnehmen dem Definitionsmodul, daß eine einzige Prozedur exportiert wird. In diese Aufgabe teilen sich die beiden Prozeduren DiskPickDemo, die exportiert wird, und DoDialogueN, die nur innerhalb des Implementationsmoduls, also nicht nach außen, sichtbar ist. Erstere dient der Initialisierung der Eingabemaske, letztere sorgt für die Anzeige des Fensters, die Eingabe und das ordnungsgemäße Abschließen; sie ist somit für uns interessanter, weshalb wir auf sie zuerst blicken.

```
DEFINITION MODULE SAMdialogue;
  PROCEDURE DiskPickDemo();
  END SAMdialogue.

IMPLEMENTATION MODULE SAMdialogue;
  FROM SYSTEM IMPORT ADR, ADDRESS, STRING;
  FROM Dialogue IMPORT ModalDialogue, FreeDialogue, Dialogue;
  FROM DBoxLoad IMPORT GetDialogueN;
  FROM Metakey IMPORT MetaRead;
  FROM StdMetakeys IMPORT Enter;
  FROM Window IMPORT Window, CurrWin, SelectWindow;
  VAR
    globalString : STRING[255];
  PROCEDURE DoDialogueN(name : STRING; a : ARRAY OF ADDRESS): BOOLEAN;
    VAR
      myDialog : Dialogue;
      goAhead : BOOLEAN;
      top : Window.Window;
    BEGIN
      top := CurrWin;
      myDialog := GetDialogueN(name,a);
      goAhead := ModalDialogue(myDialog,MetaRead)=Enter;
      FreeDialogue(myDialog, TRUE);
      SelectWindow(top);
      RETURN goAhead;
    END DoDialogueN;
  PROCEDURE DiskPickDemo();
    VAR
      newString : STRING[255];
      i, newCard, default : CARDINAL;
      myBitset : BITSET;
      parms : ARRAY [1..5] OF ADDRESS;
      pickone : CARDINAL;
    BEGIN
      pickone := 0;
      parms[1] := ADR(pickone);
      myBitset := {};
      parms[2] := ADR(myBitset);
      newString := globalString;
      parms[3] := ADR(newString);
      newCard := 0;
      parms[4] := ADR(newCard);
      IF DoDialogueN("SamDialog", parms) THEN
        globalString := newString;
      END;
    END DiskPickDemo;
  BEGIN
    globalString := "hello";
  END SAMdialogue.
```

Bild 1. Ein Modul für das Gestalten von Dialogen

```
Dialogue: "SamDialog"
Window(1, 2, 70, 15, "Disk Dialogue")
Pickone(1, 1, "PickOne: Fixed Scientific Currency Percent , +/- General Date Text")
Pickmany(14, 1, "Pickmany: Hello Goodbye Daily")
Editfield(28, 1, "Enter range name: ", 40, 16)
Editcard(14, 8, "Column Width: ", 1, 72)

End
```

Bild 2. Die Definition des Beispiel-Dialoges

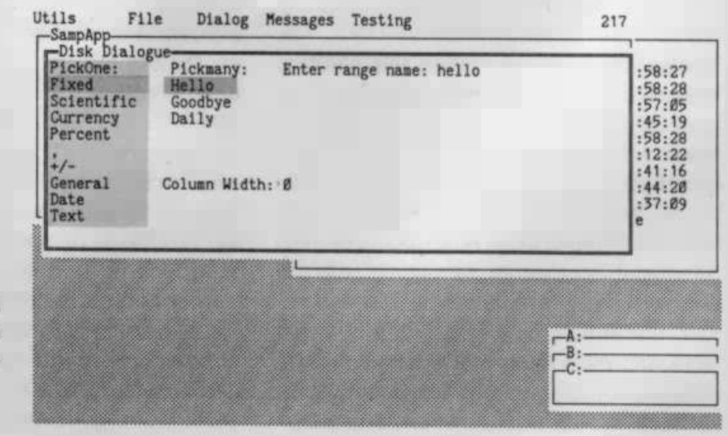


Bild 3. Das Ergebnis eines Programmlaufes

Was diese Serie bietet

In dieser Serie werden anhand der Farsight Modula-Toolbox einige interessante Beispiele aus diesem Betriebssystem geschildert. Im ersten Teil wird anhand einer Ein-/Ausgabe eine einfache praktische Anwendung des Betriebssystems vorgestellt. Der zweite Teil befaßt sich mit parallelen Prozessen und Programmen, sowie der Definition des Betriebssystems und der dritte Teil mit der virtuellen Speicherverwaltung. Die abgedruckten Programme lassen sich am Schluß zu einem Ganzen zusammenfügen. Es ist also sinnvoll, alle drei Teile aufzubewahren, um später die Zusammenhänge besser studieren zu können. Die Farsight Modula-Toolbox ist Kernstück des integrierten Paketes Farsight von Interface Technologies, Houston. Sie wurde mit dem Modula-2-Entwicklungssystem M2SDS derselben Firma entwickelt. Gegen einen Unkostenbeitrag von 10 DM (Selbstkosten) ist das vollständige Beispiel, von dem hier nur Ausschnitte abgedruckt werden können, sowie eine ausführliche Dokumentation bei folgender Adresse erhältlich: A+L AG, Im Späten 23, CH-8906 Bonstetten. Wenn Sie Applikationen planen, die in Farsight eingebunden werden können, dann können Sie die ganze Toolbox für die Entwicklung zur Verfügung gestellt bekommen. Voraussetzung ist, daß die geplanten Arbeiten gegenüber A+L genau beschrieben werden.

DoDialogueN

Am Kopf der Prozedur erkennen wir sie als Funktionsprozedur, die mit der zurückgegebenen BOOLEAN-Variable dem aufzurufenden Programm mitteilt, ob die Eingabe korrekt vonstatten gegangen ist. Als erstes wird der lokalen Variable top der Wert von CurrWin zugewiesen, das wir aus dem Modul Window des Betriebssystems importiert haben und auf das momentan oberste Fenster zeigt. Da wir dieses nach dem Dialog wieder aktivieren wollen, müssen wir den Wert speichern, bevor wir das Dialogfenster öffnen. myDialog weisen wir mit der Funktion GetDialogue einen Dialog zu, den wir unter dem Namen name im sogenannten Resourcefile finden. a ist eine Kette (Array) von Adressen, die in der gleichen Reihenfolge, wie der Dialog aufgebaut ist, auf die zugehörigen Eingabevariablen zeigen, welche durch den Dialog verändert werden sollen. Da im Prozedurkopf keine Angabe zu finden ist, wieviele Elemente die Kette aufweist, handelt es sich hier um einen offenen Array, dessen Länge erst während des Programmlaufes bestimmt wird.
Als nächsten Schritt weisen wir goAhead den Wert des gegenüberliegenden Vergleichs zu. Dieser liefert TRUE (wahr), wenn am Ende des Dialogs mit dem Benutzer Enter gedrückt wurde (bei Farsight dient Enter zur Bestätigung einer Eingabe, Esc zur Annulierung). ModalDialogue setzt sämtliche Schritte in Bewegung, die zur Anzeige des Fensters, Bearbeiten der Felder und Abschließen der Eingabe nötig sind. Der erste Parameter, myDialog, übergibt den Dialog, den das Programm gerade geladen hat, der zweite bestimmt die Eingabeprozedur. Als zweiten Parameter (ein Prozedurparameter, wie der geübte Programmierer wohl ahnt) verwenden wir die vom Betriebssystem zur Verfügung gestellte Standardprozedur aus dem Modul MetaKey, wir könnten jedoch ohne weiteres eine selbstgeschriebene hier angeben und damit das Verhalten des Dialogs genaustens bestimmen.

Am Schluß müssen wir bloß wieder den Platz freigeben, der durch den Dialog im Speicher belegt wurde und das zuletzt aktive Fenster wieder neu aktivieren. Es ist ohne weiteres möglich, daß dieses Fenster vom Dialog überdeckt wurde. Es wird dann automatisch neu gezeichnet, zusammen mit allen betroffenen anderen Fenstern. Zuletzt teilen wir dem aufrufenden Programm noch mit, ob der Dialog mit Enter abgeschlossen wurde oder nicht, also ob die neuen Daten gültig sind.

Gedanken zu DoDialogueN

Diese Funktionsprozedur ist ein Musterbeispiel für eine wiederverwendbare Funktion, insbesondere der offene Array erlaubt es, Dialoge mit unterschiedlich vielen Feldern mit dieser einen Prozedur aufzurufen. Man wird sie also sinnvollerweise nicht nur für diesen einen Dialog brauchen können, sondern in einem Modul unterbringen, von dem aus sie auch von anderen Programmen aus erreichbar (importierbar) ist. Wenn Sie Lust haben, überlegen Sie sich vielleicht noch, ob es für Ihre Bedürfnisse nötig wäre, ihr als weiteren Parameter die Leseprozedur mitzugeben, die als zweiter Parameter der Funktion ModalDialogue übergeben wird. Wie der offene Array gibt die Möglichkeit, in Modula-2 Prozedurparameter übergeben zu können, ungeahnte Flexibilität in die Hände des Programmierers. Leider sind solche Konzepte heute noch fast unbekannt.

Ein Dialog auf Disk

Bevor wir uns die Prozedur DiskPickDemo genauer ansehen, wollen wir anhand der Bilder 2 und 3 den Dialog näher betrachten. Dazu einige Vorbemerkungen zum Farsight Resourcefile, auf den wir im nächsten Teil genauer eingehen wollen. In einer speziellen Datei, die bei der Integration einer neuen Applikation in Farsight als Schnittstelle zu den schon eingebundenen Programmen dient, können unter anderem auch beliebig viele Dialoge definiert und gespeichert werden. Dies hat gegenüber fest codierten Eingabefeldern und -masken vor allem den Vorteil, daß sie leicht in andere (gesprochene) Sprachen zu übersetzen sind; dazu benötigen sie keinen zusätzlichen Platz im Programm, da sie erst während der Laufzeit geladen werden können. Die erste Zeile definiert den Namen, unter dem der Dialog aufgerufen werden kann. Gleichzeitig wird damit festgelegt, daß es sich überhaupt um einen Dialog handelt. In der zweiten Zeile wird die Größe und Position des Fensters und dessen Titel vorgegeben, in dem der Dialog vor sich gehen soll. Die beiden ersten Zahlen nennen die x,y-Werte der linken oberen Ecke, das folgende Paar der rechten unteren Ecke und der letzte Parameter enthält den Titel des Fensters, der natürlich auch leer sein darf (""). Die nun folgenden vier Zeilen demonstrieren vier mögliche Eingabearten, die für die Kommunikation mit dem Benutzer dienen. Ihre Reihenfolge muß mit derjenigen der verschiedenen Variablen übereinstimmen, auf die die Elemente des Arrays a in DoDialogueN zeigen. Das einleitende Zahlenpaar legt jeweils die obere linke Ecke des Eingabefeldes fest. Der Definitionsblock wird wie in Modula-2 üblich mit End abgeschlossen.

DiskPickDemo

Zu jeder Dialogzeile haben wir eine Variable definiert, die nun, eine um die andere, initialisiert wird. Wenn pickone nicht auf 0, sondern auf 3 gesetzt würde, dann befände sich der Balken in Bild 3 nicht auf dem Feld Fixed sondern auf Percent. Der Adresse parms[1], dem ersten Element des Adreßarray, den wir später DoDialogueN übergeben wollen, weisen wir die Anfangsadresse der Variable pickone zu. Wenn der Dialog abgeschlossen sein wird, wird letzte-

re die aktuelle Position des Menübalkens anzeigen. Bei der zweiten Eingabe kann nicht nur ein einziges Feld aus mehreren gewählt werden, sondern mehrere oder auch keine. Wenn der Benutzer den Balken auf eine der Zeilen stellt und die Markiertaste (F7) betätigt, wird das entsprechende Bit im BITSET invertiert, d. h. gesetzt, wenn noch keine Markierung vorhanden war oder andernfalls gelöscht; optisch erscheint auf dem Bildschirm ein Sternchen vor der markierten Zeile. Als nächstes übergeben wir einen vorbelegten String, der als Vorgabe im Eingabefeld angezeigt wird. Obschon der String eine Länge von maximal 40 Zeichen aufweisen darf (siehe Bild 2), werden nur 16 Zeichen angezeigt; bei der Eingabe wird der Inhalt selbsttätig, wenn nötig, hin und her geschoben („gescrollt"). Auch bei der Eingabe der Ganzzahl wird geprüft, ob der Wert im gültigen Bereich liegt, d. h. wenn wir in Bild 3 den Wert nicht ändern – 0 liegt nicht zwischen 1 und 72 –, können wir nicht über dieses Feld hinausgehen. Das Betriebssystem errechnet sich übrigens selbst die Breite des letzten Feldes, da es ja den Platzbedarf für die breiteste Zahl kennt.

Was übrig bleibt, ist rasch kommentiert: Wir rufen DoDialogueN mit dem Namen des Dialogs und den initialisierten Parametern auf und weisen, wenn nötig, die neuen Werte den entsprechenden Variablen zu, sofern der Dialog korrekt, d. h. mit Enter abgeschlossen wurde und somit die Daten gültig sind, andernfalls könnte in einem ELSE-Teil eine Reaktion programmiert werden. Falls z. B. unbedingt eine Eingabe erwartet wird, die mit Enter abgeschlossen werden muß, kann dies etwa wie folgt programmiert werden:

```
WHILE NOT DoDialogueN(„SamDialog",
              parms) DO
              END;
```

Wenn wir übrigens das nächste Mal die Prozedur DiskPickDemo aufrufen, wird im dritten Feld der String der letzten gültigen Eingabe angezeigt werden.

Gedanken zur Ein-/Ausgabe

Wer heute mit der Eisenbahn fährt, kommt wesentlich schneller voran, als vor vielleicht dreißig Jahren, und trotzdem brauchen wir nicht auf den Komfort unserer modernen Gesellschaft zu verzichten, ganz im Gegenteil: so komfortabel wie heute hatten wir es noch nie! Solche Gedanken mögen Ihnen durch den Kopf gehen, wenn Sie das eben beschriebene Beispiel studiert haben. Es ist wirklich erstaunlich, wie man dank eines guten Betriebssystems mit wenigen Zeilen eine so komplette, komfortable Eingabe zustande bringt. Als zusätzliche Information sei hier noch angemerkt, daß die Farsight Modula-Toolbox es auch erlaubt, Dialoge direkt zu programmieren, z. B. besonders einfache oder speziell komplizierte. Dies ist nicht unbedingt selbstverständlich, da oftmals bei vergleichbaren Werkzeugen vor lauter Automatismen das Beeinflussen des einzelnen Details und damit vielleicht gerade der besondere Reiz verloren gehen kann. □

Der folgende zweite Teil befaßt sich mit parallelen Prozessen und Programmen und geht in diesem Zusammenhang detaillierter auf die Gleichzeitigkeit und den Datenaustausch von Modula-Programmen ein. Er erscheint in mc, Ausgabe 11, am 24. Oktober 1988.

Excel für HP-NewWave

Microsoft und Hewlett-Packard haben eine Vereinbarung getroffen, nach der die beiden Unternehmen gemeinsam das Tabellen-Kalkulationsprogramm Microsoft Excel für die HP-NewWave-Umgebung entwickeln und vermarkten wollen.
HP-NewWave ist eine Software-Applikations-Umgebung, die auf der grafischen Bedienungsoberfläche Microsoft Windows, Version 2.0, basiert. HP-NewWave verbessert darüber hinaus die PC-Integration, indem das Programm dem Anwender einen sofortigen Überblick über die gesamten Informationsressourcen in einem Netzwerk gibt.
Die Verbesserungen in bezug auf HP-NewWave werden nach einer Darstellung von Microsoft auch in das Microsoft Excel Standard-Produkt integriert, so daß Anwender keine spezielle Version erwerben müssen, wenn sie mit HP-NewWave arbeiten. Die neue Version von Microsoft Excel wird den PC-Anwendern Zugriff auf die wichtigsten Funktionen von HP-NewWave, einschließlich der HP-Objekt-Management-Funktion und „Agents", einer Funktion zur Automatisierung systemweiter Tasks, verschaffen. Die neue Microsoft Excel-Version wird darüber hinaus eine Plattform zur Implementierung von KI-Funktionen („Künstliche Intelligenz") sein.
Hewlett-Packard plant, eine Vertriebslizenz für Microsoft Excel zu erwerben, so daß der Verkauf auch über den Vertrieb von HP erfolgt.
Hewlett-Packard und Microsoft haben die Absicht, jeweils vergleichbare Funktionen für die zukünftigen Versionen von Microsoft Excel und HP-NewWave zu entwikkeln, die auf dem Microsoft-Multitasking-Betriebssystem OS/2 und der grafischen Bedieneroberfläche Presentation Manager laufen. □

Controller von Western Digital

Gleich drei Controller für Winchester- und einen für Floppy-Laufwerke stellte Western Digital jetzt vor.
Der WDXT-GEN ist ein IBM-XT-kompatibler Winchester-Controller, der bis zu zwei Festplatten mit einem BIOS-ROM zur dynamischen Konfiguration koppelt. Auf dem Board des Halb-Slot-Controllers sind alle Empfänger und Treiber enthalten.
Mit einer 56-Bit-Fehlerkorrektur wartet der Controller WD1002a-27X auf. Er kann bis zu 50 Prozent mehr Daten auf einer ST506-Festplatte unterbringen. Eine Übersetzungsoption von 17 zu 26 Sektoren pro Spur ist eingebaut. Das BIOS erlaubt das Formatieren von schlechten Spuren, dynamische Konfiguration und virtuelle Laufwerksteilung.
Mit dem WD1003S-WA4 kommt ein multifunktionaler Disk-Controller für PC/AT auf den Markt. Er bietet in einem einzigen Slot die Controller für Winchester- und Disketten-Laufwerke sowie serielle und parallele Schnittstellen. Damit können bis zu zwei ST506/ST412-Festplatten kontrolliert werden. Der Baustein akzeptiert Datenraten bis zu 5 MBit/sek und unterstützt Laufwerke mit maximal 16 Schreib-/Leseköpfen und 2048 Zylindern.
Der Disketten-Controller WDX1002A-FOX ist kompatibel zu PC-/XT- und AT-Rechnern. Er unterstützt ein 1,44-MByte-Laufwerk sowie bis zu vier einzelne oder zwei Dual-Floppy-Laufwerke. □

Heribert Cebulla

So funktioniert die VGA

Teil 1: Ein erster Blick auf die VGA

Eines der wesentlichen Elemente der PS/2-Serie ist die neue Grafik-Schnittstelle Video Graphics Array (VGA). Unsere neue Serie macht Sie mit der VGA vertraut. Im ersten Teil beschreiben wir den Aufbau und die Funktion der VGA. In den weiteren Folgen gehen wir ausführlich auf ihre Programmierung ein.

Die VGA ist die konsequente Weiterentwicklung der EGA (Enhanced Graphics Adapter). Sie ist von der Programmseite her gesehen kompatibel zur EGA. IBM bietet sie in zwei Versionen an. Bei den Modellen IBM PS/2 50/60/70 und 80 ist die VGA auf der CPU-Platine fest eingebaut und bei dem IBM PS/2 Modell 30 und alle anderen PC und AT als normale PC-Steckkarte ausgeführt. Wie nicht anders zu erwarten, sind bereits die ersten kompatiblen VGA-Karten von anderen Herstellern am Markt. Um auch VGA-kompatible Karten in den Beitrag einzubeziehen, habe ich versucht, technische Unterlagen des jeweiligen Herstellers über Händler zu beziehen. Das Ergebnis war unbefriedigend. Die meisten Händlern waren, aus mir verständlichen Gründen, kaum in der Lage, etwas über die Funktion der VGA-kompatiblen Karte zu sagen. Die Frage nach entsprechender Dokumentation über die Karte wurde immer negativ beantwortet. Mit der Installationsanweisung und dem BIOS-Interface wäre ich nicht in der Lage gewesen, mit einem Programm auch nur ein Register direkt anzusprechen. Ich habe dann versucht, über die Vertretung der Hersteller die Dokumentation zu erhalten. Fehlanzeige! Der beste Tip einer Vertretung war, mich doch an die IBM zu wenden, die müßten so etwas ja haben, und ihre Karte wäre ja „voll" kompatibel. Oder ich könnte mir die EGA-Dokumentation beschaffen, die sei so ähnlich, aber die hätten sie auch nicht da. Warum nun diese Dokumentation von den Herstellern kompatibler Karten nicht zu erhalten ist, ist mir ein Rätsel.
Aufgrund dieser Erfahrungen befaßt sich die Serie ausschließlich mit dem Bildschirmadapter (VGA) des IBM PS/2, der Modelle 50/60/70 und 80. Nicht behandelt wird die MCGA des Modell 30.

VGA-Bildschirme

Viele Informationen zur VGA kann man den in der Literaturliste aufgeführten Büchern entnehmen [1, 2].
IBM bietet vier Bildschirme an, die für die VGA geeignet sind:

- Monochrom-Bildschirm 8503
- Farbbildschirm 8512
- Farbbildschirm 8513
- Farbbildschirm 8514.

Ihre technischen Daten sind in *Tabelle 1* zusammengefaßt. Allen vier Bildschirmen ist die Horizontal-Frequenz von 31,5 kHz, die Analog-Signaleingänge und die quadratische Auflösung gemeinsam. Der niedrigste Wert der vertikalen Ablenkfrequenz, mit der die VGA arbeiten kann, ist 50 Hz. Bei einem Bildschirm mit 50 Hz vertikaler Ablenkfrequenz könnte man bei entsprechender Programmierung der VGA eine maximale vertikale Auflösung von etwa 576 Zeilen pro Bild erreichen.

Feste Horizontal-Frequenz

Will man an die VGA einen Bildschirm eines anderen Herstellers anschließen, sollte man sich vergewissern, daß er den genannten technischen Anforderungen entspricht.
Da die Horizontal-Frequenz bei den unterschiedlichen Video-Modi konstant ist, kann man an die VGA einen Festfrequenz-Monitor anschließen, der wesentlich billiger als ein Mehrfrequenz-Bildschirm (NEC Multisync, Sony Multiscan, JVC APS) ist. Nur noch der vertikale Synchronisations-Eingang variiert in der Frequenz. Die Steilheit der vertikalen Ablenkung im Bildschirm wird über die Polarität der beiden Synchronisations-Signale (horizontal/vertikal) gesteuert (*Tabelle 2*).

Analog-Eingänge

768 Zeilen pro Bild kann jedoch nur der IBM 8514 Bildschirm in Verbindung mit dem 8514/A-Adapter darstellen.
Während die sechs digitalen Farbsignale der EGA noch TTL-Pegel hatten, sind bei der VGA nur noch drei analoge Farbsignale, nämlich Rot, Grün und Blau (RGB) vorhanden. Die Höhe der Spannung der Farbsignale liegt je nach Intensität der jeweiligen Farbe zwischen 0 Volt und 0,7 Volt. Um die von der VGA erzeugten 262144 Farben digital zu übertragen, müßte man 18 Farbleitungen zum Bildschirm führen. Außerdem ist die Pixel-Frequenz erheblich größer geworden. So ist ein Impuls für einen Bildpunkt nur noch etwa 39 ns lang. Aus diesen Gründen darf man keinen EGA-Bildschirm an der VGA-Karte anschließen.

Quadratische Auflösung

Alle IBM Bildschirme haben ein einheitliches Höhen-/Seitenverhältnis, d. h. daß bei den VGA Modi 11H und 12H (640 × 480) der Abstand zwischen den Bildpunkten horizontal wie vertikal immer gleich groß ist. Ein Viereck von 10 × 10 Bildpunkten ist auf allen Bildschirmen wirklich ein Quadrat. Ebenso ist ein Kreis immer ein Kreis und nicht manchmal eine Ellipse. Der von der Turbo Graphics Toolbox bekannte „Aspect Faktor" fällt damit fort, oder ist 1.

IBM PS/2 VGA

Die gesamte Schaltung der VGA befindet sich auf der CPU-Platine der IBM PS/2 Modelle 50/60/70 und 80. Sie besteht im wesentlichen nur aus dem VGA-Chip, dem Video-DAC und dem 256 KByte großen Bildspeicher.

Die Video-Schnittstelle ist auf der CPU-Platine direkt am Mikrokanal angeschlossen.
Bis auf den Bildspeicher enthält der VGA-Chip alle Funktionen, die bisher auch auf der EGA vorhanden waren. Zwischen VGA-Chip und Bildschirm ist der Video-DAC (Digital Analog Converter) geschaltet, der die drei analogen Ausgangssignale (RGB) erzeugt. Die horizontale Ablenkfrequenz liegt mit 31,5 kHz (durch das BIOS) fest. Der Bildspeicher ist 256 KByte groß und besteht aus 4 Speicherblöcken von je 64 K × 8 Bit. Alle bisherigen von IBM geschaffenen Bildschirmmodi werden von der VGA un-

Suchen sie einen Partner mit dem Sie wachsen können?

Die INTRA-führender Hersteller von Monitoren - bietet seit Jahren Ihren Kunden eine stetig wachsende Anzahl von Anzeigegeräten und Displays.
Von der Bildröhre bis zum 20" Farb-Videospielmonitor, vom Composite-Monitor bis zum IBM-kompatiblen Monochrom-Monitor und 14" CGA/EGA Multi-SYNC-Bildschirm.
Wir haben nicht nur das Wissen und Können, sondern auch die Qualität.
Wir wollen expandieren und weiterhin qualitativ hochwertige HiTech entwickeln - schon heute fertigen wir bis zu 30,000 Bildschirme monatlich.
Mit 3, 5 Mio Dollar Kapital, 250 Mitarbeitern und einem Werksgelände von knapp 6,000 qm haben wir die besten Voraussetzungen für Wachstum.
Unsere Zukunftspläne erfordern eine verstärkte Präsenz in den Bereichen Ganzseitenmonitor sowie eine Ausweitung der Desk-top-, Modem- und Lap-top-Technologien.
Sie sind in diesem Bereich tätig, verfügen über ausreichende Erfahrung und wollen kooperieren?
Dann sollten Sie mit uns Kontakt aufnehmen. Es wird sich für Sie lohnen.

"VGA" MONOCHROME MONITOR
MODEL 14HP34V
1. 14" FLAT SCREEN
2. PS/2, VGA COMPATIBLE (31.5KHZ) ANALOG VIDEO
3. CRT: PAPER-WHITE, AMBER,GREEN

14" MONOCHROME MONITOR
MODEL 14HP33T
1. 14" FLAT SCREEN
2. DUAL FREQ (15.75/18.432KHz)
3. CRT: PAPER-WHITE, AMBER,GREEN
4. WITH REVERSE SWITCH

14" HIGH-RESOLUTION COLOR MONITOR
MODEL
14CH113 (EGA 640 X 350)
14CH114 (CGA 640 X 200)
14CH115 (MULTI-SYNC 800 X 600)
14CH116 (VGA 640 X 350
640 X 400
640 X 480)

• "VGA" MONOCHROME MONITOR





• 14" MONOCHROME MONITOR

• 14" HIGH-RESOLUTION COLOR MONITOR

Intra Electronics Co., Ltd.
Room No. 618, 6th Fl., 9, Lane 3,
Min Sheng West Rd., Taipei, Taiwan, R.O.C.
Tel: (02)597-7027 Tlx: 19925 INTRA Fax: 886-2-5418513

VED.FCC APPROVED
• OEMs WELCOME !!

Tabelle 1: Technische Daten der VGA-Bildschirme von IBM

IBM Monochrom-Bildschirm 8503	
Bildschirmdiagonale	12 Zoll (30,5 cm)
Bildschirm-Farbe	Perlweiß
Signaleingang	analog
Horizontale Frequenz	31,5 kHz
Vertikale Frequenz	50...70 Hz
Auflösung:	88 Punkte/Zoll (3,46 Punkte/mm)
Text	640/720 × 400
Grafik	640 × 480, 640 × 400, 640 × 350
Farbskala	unendlich viele Grautöne
mit VGA	64 Grautöne
IBM Farbbildschirm 8512	
Bildschirmdiagonale	14 Zoll (35,5 cm)
Punktabstand	0,41 mm (Schlitzmaske)
Signaleingang	analog RGB
Horizontale Frequenz	31,5 kHz
Vertikale Frequenz	50...70 Hz
Auflösung:	68 Punkte/Zoll (2,67 Punkte/mm)
Text	640/720 × 400
Grafik	640 × 480, 640 × 400, 640 × 350
Farbskala:	unendlich
mit VGA	2/16/256 Farben aus 262 144 Farben
IBM Farbbildschirm 8513	
Bildschirmdiagonale	12 Zoll (30,5 cm)
Punktabstand	0,28 mm (Lochmaske)
Signaleingang	analog RGB
Horizontale Frequenz	31,5 kHz
Vertikale Frequenz	50...70 Hz
Auflösung:	88 Punkte/Zoll (3,46 Punkte/mm)
Text	640/720 × 400
Grafik	640 × 480, 640 × 400, 640 × 350
Farbskala:	unendlich
mit VGA	2/16/256 Farben aus 262 144 Farben
IBM Farbbildschirm 8514	
Bildschirmdiagonale	16 Zoll (40,6 cm)
Punktabstand	0,31 mm (Lochmaske)
Signaleingang	analog RGB
Horizontale Frequenz	
ohne Zeilensprung	31,5 kHz
mit Zeilensprung	35,52 kHz
Vertikale Frequenz	43,48...70 Hz
Auflösung:	92 Punkte/Zoll (3,62 Punkte/mm)
Text	640/720 × 400
Grafik	640 × 480, 640 × 400, 640 × 350 1024 × 768 (IBM 8514/Adapter)
Farbskala:	unendlich
mit VGA	2/16/256 Farben aus 262 144 Farben

Tabelle 2: Polarität der Synchronisations-Signale [1]

horizontal	vertikal	Zeilen/Bild
+	–	350
–	+	400
–	–	480
+	+	768*

*) nicht mit VGA

terstützt. Dazu gehören MDA (Monochrome Display Adapter), CGA (Color Graphics Adapter) und EGA (Enhanced Graphics Adapter). Die Hercules-Karte wird von der IBM VGA nicht emuliert.
Bei der CGA-Emulation werden alle 200 Zeilen Modi doppelt auf dem Bildschirm dargestellt (double scan). Ein Bild mit einer Auflösung von 200 Zeilen wird auf dem Bildschirm mit 400 Zeilen abgebildet. Diese Darstellungsart ist bei alten CGA-Grafik-Programmen für das Auge angenehm, da jetzt die störenden Unterbrechungen der Bildstruktur in der Senkrechten verschwinden.
Alle VGA-BIOS-Modi (+: vergrößerten Modi mit 400 Zeilen; *: vergrößerten EGA-Modi) sind in *Tabelle 3* zusammengefaßt.
Die Spalte Bildrand (Border) gibt an, ob die Farbe des Bildschirmrandes bei dem entsprechenden BIOS-Modus programmierbar ist. Die Farbe des Bildschirmrandes ist frei wählbar und bei allen 80-Spaltenmodi und auch beim BIOS-Modus 13 genau eine Zeichenbox breit. Obwohl die Tabelle 3 für alle 40-Spaltenmodi keinen Bildrand angibt, ist hier die Farbe des Bildschirmrandes doch programmierbar, jedoch ist die Randbreite hier nicht symmetrisch. Der vertikale Bildrand ist eine 40-Spalten Zeichenbox breit, während der horizontale Bildrand nur halb so hoch (eine 80-Spalten Zeichenbox) ist.

Der VGA-Chip

Der VGA-Chip besteht im wesentlichen aus folgenden fünf Funktionseinheiten (*Bild 1*):

- Bildschirm-Controller (CRTC)
- Sequencer (SEQ)
- Grafik-Controller (Graph Cntl)
- Attribut-Controller (Attrib Cntl)
- Adreß/Daten-Multiplexer (MUX).

Die insgesamt 64 Register der VGA sind in sechs Registergruppen zusammengefaßt. Sie lassen sich über I/O-Befehle programmieren. Während bei der EGA noch alle Register nur beschreibbare Register (write only register) waren, sind im VGA-Chip alle Register auch lesbar.

Bildschirm-Controller

Die wesentliche Aufgabe des Bildschirm-Controllers (CRTC) ist die Erzeugung der horizontalen/vertikalen Synchronisations-Signale für die unterschiedlichen VGA-BIOS-Modi. Außerdem erfolgt vom CRTC die Adressierung und der Refresh des dynamischen Bildschirmspeichers. Er generiert auch das Zeitverhalten des Cursors und das Unterstreichen von Text. Auf die Bedeutung einiger seiner 24 Register gehe ich während der Serie ein.

Sequenzer

Der Sequenzer generiert die Signale für den dynamischen Bildschirmspeicher (z. B. RAS und CAS) und bestimmt für den Bildrefresh das Zeitverhalten der Zeichen. Eine weitere Funktion des Sequenzers ist die Kontrolle über die periodische Verteilung der Bildspeicherzugriffe, durch den CRTC für den Bildrefresh, und durch den Mikroprozessor. Mit dem Map-Mask-Register des Sequenzers wählt man die Bildspeicherebene aus, in die geschrieben werden soll. Auf die wichtigsten Funktionen der fünf Register im Sequenzer gehe ich später näher ein.

Grafik-Controller

Der Grafik-Controller (*Bild 2*) ist neben dem CRTC der eigentliche Prozessor des VGA-Adapters. Er ist die Schaltzentrale zwischen dem Bildspeicher und dem Attri-

Tabelle 3: Alle BIOS-Modi der VGA auf einen Blick [2]

Modus	Typ	Farben	Alpha-Format	Startadresse	Box-Größe	Max. Seiten	Vertikale Frequ.	Auflösung	Double Scan	Farbrand
0, 1	A/N	16/256K	40 × 25	B8000	8 × 8	8	70 Hz	320 × 200	ja	nein
2, 3	A/N	16/256K	80 × 25	B8000	8 × 8	8	70 Hz	640 × 200	ja	ja
0*, 1*	A/N	16/256K	40 × 25	B8000	8 × 14	8	70 Hz	320 × 350	nein	nein
2*, 3*	A/N	16/256K	80 × 25	B8000	8 × 14	8	70 Hz	640 × 350	nein	ja
0+, 1+	A/N	16/256K	40 × 25	B8000	9 × 16	8	70 Hz	360 × 400	nein	nein
2+, 3+	A/N	16/256K	80 × 25	B8000	9 × 16	8	70 Hz	720 × 400	nein	ja
4, 5	APA	4/256K	40 × 25	B8000	8 × 8	1	70 Hz	320 × 200	ja	nein
6	APA	2/256K	80 × 25	B8000	8 × 8	1	70 Hz	640 × 200	ja	ja
7	A/N	–	80 × 25	B0000	9 × 14	8	70 Hz	720 × 350	nein	ja
7+	A/N	–	80 × 25	B0000	9 × 16	8	70 Hz	720 × 400	nein	ja
D	APA	16/256K	40 × 25	A0000	8 × 8	8	70 Hz	320 × 200	ja	nein
E	APA	16/256K	80 × 25	A0000	8 × 8	4	70 Hz	640 × 200	ja	ja
F	APA	–	80 × 25	A0000	8 × 14	2	70 Hz	640 × 350	nein	ja
10	APA	16/256K	80 × 25	A0000	8 × 14	2	70 Hz	640 × 350	nein	ja
11	APA	2/256K	80 × 30	A0000	8 × 16	1	60 Hz	640 × 480	nein	ja
12	APA	16/256K	80 × 30	A0000	8 × 16	1	60 Hz	640 × 480	nein	ja
13	APA	256/256K	40 × 25	A0000	8 × 8	1	70 Hz	320 × 200	ja	ja

Tabelle 4: Die Grundfunktionen des Attribut-Bytes [1]

	Attribut-Byte							
	7	6	5	4	3	2	1	0
	B/I	R Hintergrund	G	B	I/CS	R Vordergrund	G	B
Normal (weiß auf schwarz)	B/I	0	0	0	I/CS	1	1	1
Invers (schwarz auf weiß)	B/I	1	1	1	I/CS	0	0	0
keine Anzeige (schwarz)	B/I	0	0	0	I/CS	0	0	0
keine Anzeige (weiß)	B/I	1	1	1	I/CS	1	1	1
Mono = unterstrichen/Farbe = blau	B/I	0	0	0	I/CS	0	0	1

I = hervorgehoben, B = blinkender Vordergrund (Zeichen), CS = Zeichengenerator-Auswahl
Bei der Wahl eines Bildschirmmodus über das BIOS wird das Bit 7 (B/I) auf Blinken und das Bit 3 (I/CS) auf Hervorgehoben eingestellt.

but-Controller für den Bildaufbau auf dem Bildschirm einerseits, und dem Bildspeicher und Mikroprozessor für die Schreib- und Leseoperationen zum Bildspeicher andererseits.
Der Grafik-Controller führt dabei im wesentlichen folgende drei Grundfunktionen parallel aus:

- Grafik-Darstellung
- Text-Darstellung
- Logische Funktionen.

Im Grafikmodus, auch APA-Modus (APA = All Points Addressable) genannt, werden die Bildspeicherdaten aus den vier Bildspeicherebenen parallel ausgelesen (8 Bit pro Bildspeicherebene) und dann seriell zum Attribut-Controller übertragen. Bit 7 ist das erste übertragene Bit innerhalb eines Byte.
Im Textmodus, auch A/N-Modus genannt, werden die durch den Textspeicher adressierten Daten des Zeichengenerators mit den dazu gehörenden Attribut-Daten parallel zum Attribut-Controller (*Bild 2*) übertragen.

Logische Funktionen

Der Mikroprozessor kann nicht nur in den Bildspeicher schreiben und ihn auslesen, sondern auch mit vier Schreibmodi und vier logischen Funktionen die Bilddaten beeinflussen.
In zwei Lesemodi kann er Bilddaten aus dem Bildspeicher lesen. Im Lesemodus 1 kann man die Farbe eines einzelnen Bildpunktes oder bis zu acht Bildpunkte gleichzeitig mit dem Inhalt des Farbvergleichs-Registers (Color Compare Register) vergleichen. Der Grafik-Controller kann in allen Schreibmodi Bits maskieren.
Das Schreiben von 32 Bit (ein Byte in allen 4 Speicherebenen Map 0..3) in einem Speicherzyklus und das Schreiben über die Bytegrenze ist ebenfalls möglich. Die detaillierte Beschreibung erfolgt im Rahmen dieser Serie.

Attribut-Controller

Die Hauptaufgabe des Attribut-Controllers (*Bild 3*) besteht darin, die durch den CRT-Controller adressierten Bildspeicherdaten so aufzubereiten, daß sie mit der gewählten Farbe am Bildschirm erscheinen. Vom Bildspeicher gelangen die Daten über den Grafik-Controller an den Eingang des Attribut-Controllers. Im Grafik-Modus werden die Daten bereits vom Grafik-Controller serialisiert, während beim Textmodus die Daten eines Zeichens erst vom Attribut-Controller serialisiert und mit den Attributdaten verknüpft werden. Am Ausgang des Attribut-Controllers führt ein 8-Bit breiter Bus (P0...P7) zum Video-DAC (Video Digital Analog Converter). Eine auf diesem Bus anstehende Adresse wählt eine der 256 Farben (aus 262144) für den jeweils aktuellen Bildpunkt aus.
Nebenbei steuert der Attribut-Controller noch die Funktionen Blinken, Unterstreichen, Cursor und das „PEL panning" (PEL: picture element). Die Beschreibung der 20 Register des Attribut-Controller erfolgt später.

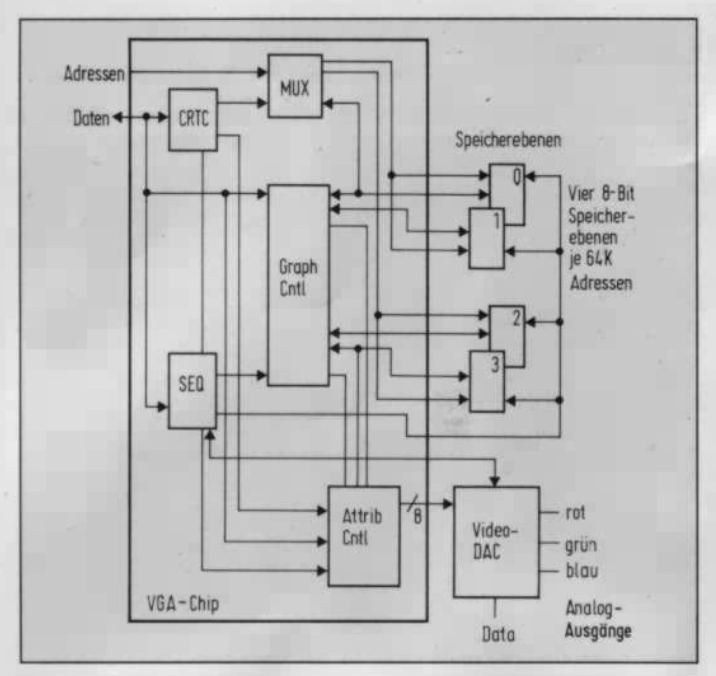


Bild 1. Blockschaltbild der VGA-Karte [1]

Bild 2. Blockschaltbild des Grafik-Controllers [1]

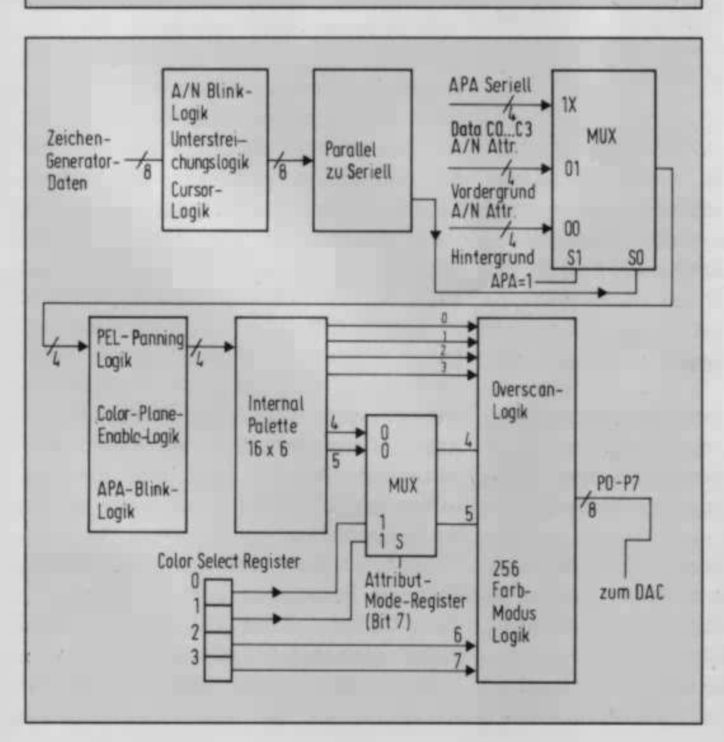


Bild 3. Blockschaltbild des Attribut-Controllers (der 256 Farbmodus ist hier nur angedeutet) [1]

Adreß-/Daten-Multiplexer

Der Bildspeicher wird von drei verschieden Quellen adressiert: Vom CRTC für den Bildrefresh, dem Mikroprozessor für den Bildaufbau und der Speicherebene 0 bei allen Text-Modi. Den jeweiligen Pfad der Adressierung gibt der Multiplexer (MUX) frei.

BIOS

Das BIOS stellt im wesentlichen die Basis-Programm-Unterstützung für den VGA-Adapter zur Verfügung. Es ist Teil des System-BIOS und enthält auch die Zeichengeneratoren. Da es jetzt im Gegensatz zur EGA nicht mehr byteweise adressiert wird, ist es auch erheblich schneller.

Bildspeicher und Organisation

Der Bildspeicher ist 256 KByte groß und in vier Speicherebenen (Map 0, Map 1, Map 2 und Map 3) zu je 64 KByte aufgeteilt. Alle vier Speicherebenen werden parallel adressiert und belegen daher nur 64 K Adressen. Der Bildspeicher wird grundsätzlich byteweise adressiert, was sehr häufig als negativ angesehen wird, in Wirklichkeit aber sinnvoll ist. Nur im reinen Textmodus wäre eine wortweise Adreßierung (2 Byte) schneller. Im Grafikmodus wäre eine wortweise Adreßierung sogar hinderlich und langsamer, wenn z. B. das Bild im Bildschirmspeicher erzeugt oder manipuliert wird. Jeder Bildspeicherebene ist ein 8 Bit breites Register zugeordnet. Diese vier Register des Bildspeichers werden auch manchmal Mikroprozessor-Latches genannt, da man mit ihrer Hilfe auch logische Operationen durchführen kann. Wegen dieser Anordnung kann der Prozessor 32-Bit-Schreib- oder Leseoperationen in einem Speicherzyklus ausführen. Das ist von besonderem Wert, wenn man z. B. Daten im Bildspeicher bewegen will.

VGA-Operationsmodi

Grundsätzlich stehen zwei Arbeitsmodi zur Verfügung, der Textmodus oder auch Alphanumerik-Modus (A/N) genannt, und der Grafikmodus. Im Textmodus sind mit den grafischen Sonderzeichen nur einfache Blockgrafiken realisierbar. Dagegen ist im Grafikmodus außer der Punktgrafik mit einer maximalen Auflösung von 640 × 480 jeder beliebige Zeichenfont möglich.

VGA-Textmodus

Die VGA-Modi 0...3 und 7 sind die vom BIOS unterstützten Textmodi. Das Daten-

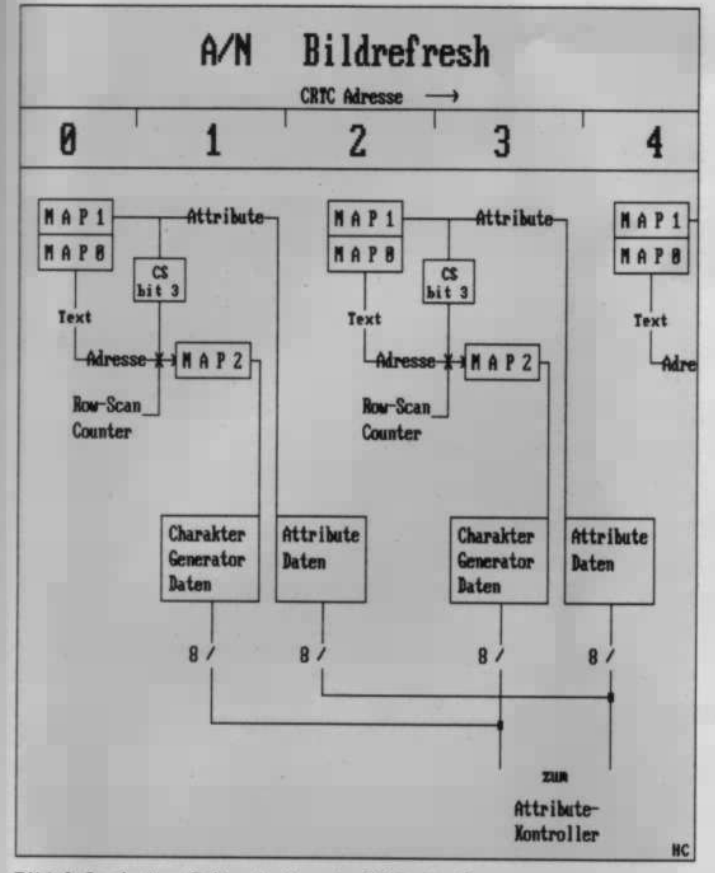


Bild 4. Logisches Zeitverhalten des Bildrefreshs

Bild 5. So speichert die VGA im Textmodus Zeichen

format für diese Textmodi ist das gleiche wie bei den alten Bildschirmadaptern MDA, CGA, und EGA. Für jedes Zeichen, das am Bildschirm angezeigt werden soll, sind zwei Byte, d. h. zwei hintereinander liegende Adressen nötig. Das Byte mit dem ASCII-Code eines Zeichens muß im Bildschirmspeicher auf einer geraden Adresse, und das dazugehörende Attribut-Byte auf der nächsten ungeraden Adresse im Bildspeicher stehen (*Bild 6*).

Die Grundfunktionen des Attribut-Byte gehen aus *Tabelle 4* hervor. Auf das Bit 3 (I/CS: Intensified/Character Select Bit) im Attribut-Byte, gehe ich später näher ein. Vom gewählten Textmodus hängt es ab, ob der Bildschirmspeicher 2 KByte oder 4 KByte pro Bildschirmseite groß ist. Es können bis zu acht Bildschirmseiten im Bildspeicher gleichzeitig gehalten werden. Zwischen den gespeicherten Bildschirmseiten kann man blitzschnell umschalten.

Wird ein Textmodus über Interrupt 10H aufgerufen, z. B. Modus 3, so kopiert das BIOS zuerst die Zeichengenerator-Daten aus dem ROM (Teil des BIOS) in die Speicherebene 2 (Map 2). Aus der Organisation der Speicherebene 2 (*Bild 7*) ist zu ersehen, daß jeder Zeichen-Generator 8 KByte groß ist, nämlich 256 verschiedene Zeichen * 32 Byte pro Zeichen. Damit können Zeichen geladen werden, die maximal 32 Rasterzeilen groß sind. In Bild 5 ist z. B. das Bitmuster für das große „A" dargestellt. Die Adressierung des Bitmusters in der Speicherebene 2 geschieht durch den hexadezimalen Wert des Zeichens selbst. So ist z. B. die dezimale Start-Adresse für das große „A" (41H) in der Speicherebene 2:
Adresse(dez) = 41H × 32 = 2080.

Die nachfolgenden Bitmuster der Zeichen werden mit einem 5-Bit-Zähler, dem sogenannten Rasterzeilenzähler, auch Row Scan Counter genannt, adressiert.

Die Zeichen (Textdaten), die am Bildschirm zu sehen sind, werden durch den Mikroprozessor in die Speicherebene 0 (Map 0), gerade Speicheradresse, und die dazugehörenden Attribut-Daten in die Speicherebene 1 (Map 1), ungerade Speicheradresse, geschrieben. Das niedrigste Mikroprozessor-Adreßbit „A0" und die Odd/Even-Steuerung der VGA-Karte bestimmen die Auswahl von Speicherebene 0 oder 1. Das Adreßbit „A0" nimmt damit nicht an der Adressierung des Bildspeichers teil. Das bedeutet nichts anderes, als daß das erste Textbyte und das erste Attributbyte im Bildschirmspeicher an der gleichen Adresse, nämlich 0000H stehen, und das zweite Text- und Attributbyte an der Adresse 0002. Die Reihenfolge, wie man in die Speicherebene 0 oder 1 schreibt, ist beliebig. Ebenso können die beiden Speicherebenen auch vom Mikroprozessor ausgelesen werden. Da in der Speicherebene 0 nur die geraden und in der Speicherebene 1 nur die ungeraden Adressen genutzt werden, ist es jetzt auch verständlich, daß der Bildschirmspeicher bei 25 Zeilen und 80 Spalten nicht 25 (Zeilen) × 80 (Spalten) = 2000 Byte groß ist, sondern 4000 Byte groß sein muß. Im Gegensatz zum Mikroprozessor lautet für den im VGA-Chip (Bild 8) integrierten CRT-Controller die erste Bildspeicher-Adresse immer 0000H.

Bildrefresh im Textmodus

Wie entsteht im Textmodus ein Bild? Im Textmodus besteht ein Bild aus 400 Rasterzeilen. Bei einer Höhe von 16 Rasterzeilen pro Zeichenmatrix können somit genau 25 Textzeilen am Bildschirm dargestellt werden. In der Speicherebene 0 (Map 0) befinden sich die Textdaten und in der Speicherebene 1 die dazu gehörenden Attributdaten. Der Zeichengenerator ist in der Speicherebene 2 geladen. Das „logische" Zeitverhalten des Bildaufbaus geht aus Bild 4 hervor. In Bild 5 ist die linke obere Ecke des Bildschirmes stark vergrößert dargestellt.

Der CRT-Controller erzeugt für den Bildrefresh zwei Adreß-Sequenzen, und zwar die CRTC-Adresse 0 1 2 3 4 und die 5-Bit Rasterzeilenzähler-Adresse (Row Scan Counter).

Die CRTC-Adresse adressiert nur die Speicherebenen 0 und 1. Im Rasterzeilenzähler wird die Anzahl der Rasterzeilen pro Zeichenbox gezählt. Ist z. B. wie in Bild 5 die Zeichenbox 16 Zeilen groß, zählt der Rasterzeilenzähler von 0 bis 15. Nach jeder Rasterzeile wird der Rasterzeilenzähler um eins erhöht. Vor jeder neuen Textzeile wird der Rasterzeilenzähler wieder auf 0 gesetzt. Am Anfang einer neuen Rasterzeile wird die CRTC-Adresse wieder auf den Anfang der entsprechenden Textzeile gesetzt.
Bild 4 zeigt das logische Zeitverhalten, wie der Bildspeicher für den Bild-Refresh ausgelesen wird.
Es wird zuerst die Speicherebene 0 und 1 (Map 0 und Map 1) ausgelesen. Mit dem ASCII-Code des Zeichens aus der Speicherebene 0, und dem Stand des Rasterzeilenzählers wird im nächsten Zyklus das erste Byte aus dem Zeichengenerator ausgelesen. Damit sind die ersten acht Bildpunkte und das zugehörige Attributbyte ausgelesen und gehen zum Attribut-Controller (Bild 3).

Die vom Zeichengenerator kommenden Daten werden vom Attribut-Controller serialisiert. Am Multiplexer wird dann abhängig von den seriellen Bits, 1 oder 0, die Vordergrund- oder die Hintergrundfarbe ausgewählt. Über die PEL-Panning-Logik gelangen die vier Bit des selektierten Farbattributs zur internen Farbpalette (Internal Palette 16 × 6). Sie besteht aus 16 Registern und war bereits bei der EGA die letzte Station vor dem Bildschirm. Zwei Bit (4 und 5) der internen Farbpalette steuern zusammen mit zwei Bit des Farbauswahlregisters (Color Select Register) einen Multiplexer. Eine zusätzliche Logik verknüpft die zwei Bit am Ausgang des Multiplexers mit den vier Bit der internen Farbpalette und zwei weiteren Bit des Farbauswahlregisters. Das Ergebnis dieser Verknüpfung ist die Farbadresse P0...P7. Im Video-DAC (siehe Bild 1) adressiert sie dann eines der 256 Farbregister. Der Inhalt des selektierten Registers bestimmt das für den Bildschirm bestimmte Signal auf der Rot-, Grün- und Blau-Leitung (RGB). Damit ist der erste Bildpunkt auf dem Bildschirm zu sehen.
Dieser Vorgang in Bild 4 wiederholt sich so lange, bis in der ersten Textzeile die letzte Text-Byte-Position verarbeitet worden ist, d. h. das Ende der Rasterzeile 0 in Bild 5 erreicht ist.
Jetzt beginnt die ganze in Bild 4 dargestellte Prozedur von vorn, aber mit dem kleinen Unterschied, das der Rasterzeilenzähler um eins erhöht worden ist. In der nächsten Rasterzeile, Rasterzeile 1, wird das zweite Byte aus jeder Zeichengenerator-Box ausgelesen.

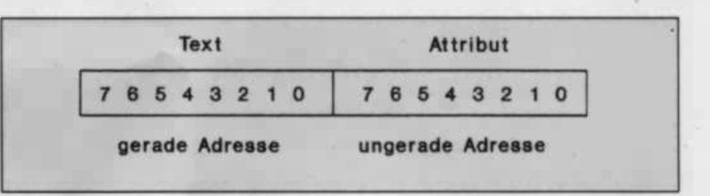


Bild 6. Ein Zeichen wird an einer geraden Adresse und sein Attribut an einer ungeraden Adresse gespeichert

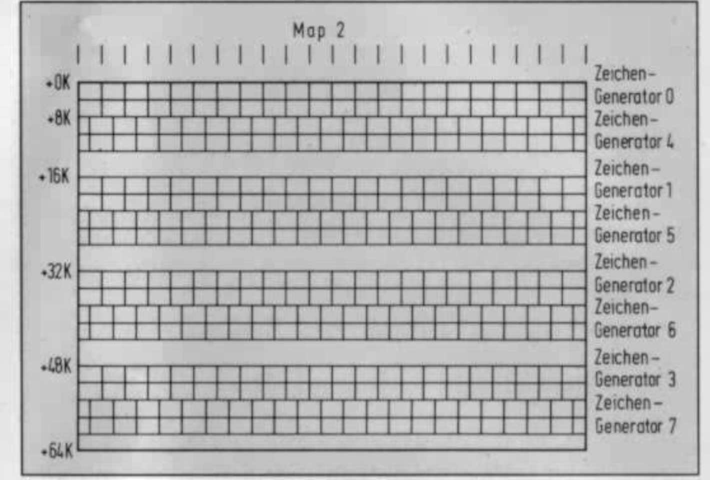


Bild 7. In der Speicherebene 2 werden die verschiedenen Zeichensätze gespeichert [1]

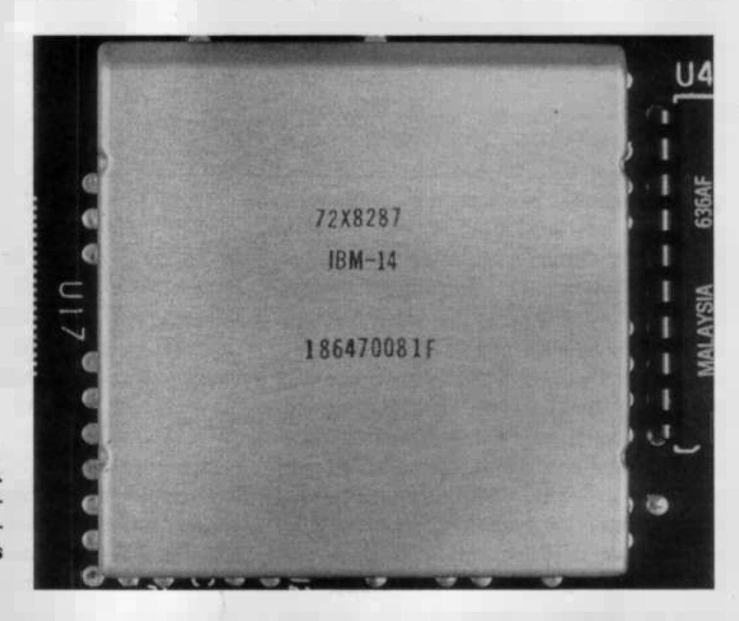


Bild 8. So sieht der Original-VGA-Baustein aus

Erst, wenn das letzte Byte aus der Zeichengenerator-Box verarbeitet worden ist, d. h. in unserem Beispiel nach 16 Rasterzeilen, wird die CRTC-Adresse (nicht wieder auf 00) auf die nächste Textzeile gesetzt. Die Prozedur beginnt von vorn bis letztlich das ganze Bild aufgebaut ist.

In allen Textmodi besteht ein Zeichen am Bildschirm nicht aus acht, sondern aus neun Bildpunkten. Der neunte Bildpunkt wird durch den Attribut-Controller hinzugefügt und ist für die ASCII-Zeichen C0H...DFH (das sind Text-Grafik-Codes) die Kopie des achten Bildpunktes. Mit diesem Trick kann man Blockgrafik-Zeichen über die Zeichenbox hinaus verbinden. Bei allen anderen ASCII-Zeichen ist der neunte Bildpunkt die Hintergrundfarbe.

Ist im Attributbyte z. B. das Blinking Bit an, werden die Zeichengenerator-Daten in Bild 3 periodisch auf Null gesetzt, bevor sie serialisiert werden. Beim Blinken wechselt am Bildschirm die Farbe eines Zeichens periodisch zwischen der Vordergrund- und Hintergrundfarbe.

Im nächsten Teil der Serie beschreibt mc das VGA-BIOS. Die Ausgabe 11 erhalten Sie ab dem 24. Oktober 1988 an Ihrem Kiosk.

Literatur

[1] IBM Personal System/2, Model 50 and 60 Technical Reference
[2] IBM Personal System/2, Seminar Proceedings
[3] Richard Wilton: Video Systems. Microsoft Press

Hagen Völzke

Fließkomma-Arithmetik und IEEE-Spezifikationen

Teil 1: Standards und Strukturen

Numerische Probleme sind eine häufige Computer-Anwendung, wobei meist nicht nur hohe Rechengeschwindigkeit gefordert, sondern auch Wert auf hohe Genauigkeit gelegt wird. Dieser Beitrag befaßt sich mit Standards und Datenstrukturen, die beim Rechnen mit Computern Verwendung finden.

Betrachten wir zu Anfang zwei einfache Ausdrücke:
x = (1 + 1e-20) – 1 und
x = (1 – 1) + 1e-20.
Anscheinend ergeben beide Ausdrücke das gleiche Ergebnis für x. Fragt man jedoch den Computer, so liefert er im ersten Fall als Resultat 0, im zweiten Fall hingegen das erwartete Ergebnis von 10^{-20}. Offenbar wurden hier fundamentale Gesetze der Mathematik verletzt: das Assoziativgesetz und das Kommutativgesetz. Lediglich durch Umstellen und andere Klammerung wird der Ausdruck vollkommen unterschiedlich berechnet, obwohl mathematisch das gleiche Ergebnis zu erwarten ist. Hier spielt ganz offensichtlich die Reihenfolge der Auswertung eine große Rolle für die Korrektheit des Ergebnisses. Eine Division durch das Ergebnis dieser Operation führt im ersten Fall zu einer Division durch Null, also einem fatalen Programmfehler, im zweiten Fall jedoch wird fehlerfrei weitergearbeitet. Derartige Fehler können also große Probleme nach sich ziehen. Dabei wurden doch nur die Operationen Subtraktion und Addition benutzt. Aber gerade die Subtraktion ist die labilste arithmetische Operation, die ein Computer durchführen kann – die komplizierteren Divisionen und Multiplikationen bereiten oft weit weniger Probleme.

Die Numerik

Die Numerik ist eines der Spezialgebiete der Mathematik, das durch die Computerära noch interessanter wurde. In diesem Fachgebiet beschäftigen sich die Mathematiker auch mit all den Problemen, Tricks und Methoden, die auftreten, wenn man dem Computer das Rechnen beibringen will. Dabei entwickelt die Numerik ihre eigenen Rechenregeln. Assoziativgesetz und Kommutativgesetz sowie einfache Umkehrregeln gelten nur noch bedingt, oftmals abhängig von den Eingabedaten, so daß sich Rechenfehler nur schwer von vornherein abwenden lassen.

Tabelle 1: Mantisse und Exponent

Die Schreibweise „1e9“ ist eine Abkürzung für 1×10^9, also für den Wert 1 000 000 000, eine Eins mit neun Nullen. Man nennt dies die Exponentialdarstellung. Im Binärsystem lautet die Exponentialdarstellung beispielsweise: „1e101“ (ist eine Abkürzung für 1×2^5). Das Zeichen „e“ wird also immer gegen die Basis des jeweiligen Zahlensystems ausgetauscht.
Die Zahl 100,0 entspricht $1{,}0 \times 10^2$ – eine Verschiebung des Kommas nach links entspricht einer Erhöhung des Exponenten.
Die Zahl 0,001 entspricht $1{,}0 \times 10^{-3}$ – eine Verschiebung des Kommas nach rechts entspricht einer Verringerung des Exponenten.

Fast alle Mikroprozessoren beherrschen zunächst nur den Umgang mit ganzen Zahlen, den sogenannten Integer-Zahlen. Doch für die täglichen Berechnungen reicht das nicht aus: Bereits beim Addieren von Geldbeträgen, wie es in jeder Buchhaltung vorkommt, benötigt man auch Bruchteile von Einheiten, eben Pfennige und manchmal sogar Bruchteile von diesen (etwa bei Benzinpreisen).
Bei Geldbeträgen könnte man nun auf die Idee kommen, immer mit Pfennigen zu rechnen, denn dann hätte man ja wieder ganze Zahlen. Bei der Ein- und Ausgabe in der Einheit „Mark“ kann man dann jeweils das Komma (bzw. den Dezimalpunkt) berücksichtigen. Diese Form der Arithmetik nennt man Fixpunktarithmetik, da mit einer festen Kommaposition gearbeitet wird. Während es im europäischen Raum üblich ist, Bruchteile von Einheiten durch ein Komma abzutrennen, wird im amerikanischen Raum ein Punkt benutzt. Die Computer, die vor allem durch amerikanische Einflüsse geprägt wurden, verwenden normalerweise den Punkt zur Trennung. „23,45 DM“ entspricht also „23.45 DM“. Wir wollen daher unter den Begriffen Fixpunkt-, Fixkomma-, Festpunkt-, und Festkomma-Arithmetik das gleiche verstehen. Während eine feste Lage des Kommas recht einfach im Computer zu handhaben ist, wird es komplizierter, wenn man das Komma verschieben will, je nachdem ob man mit besonders großen oder kleinen Zahlen rechnen möchte. Man spricht dann auch von gleitender oder fließender Kommastellung. Ein Programmpaket, das ein gleitendes Komma unterstützt, heißt im Fachjargon „floating-point“-Paket.

Die Exponential-Darstellung

Während man sich eine feste Lage des Kommas nur für die Ein- und Ausgabe merkt, muß die Lage eines verschiebbaren Kommas zusammen mit der „eigentlichen“ Zahl abgespeichert werden. Die „eigentliche“ Zahl ist dann eine Ziffernfolge, man nennt sie die Mantisse. Die Kommastellung relativ dazu wird durch den Exponenten angegeben. Unseren DM-Betrag von oben kann man also durch die Mantisse 2345 und den Exponenten –2 abspeichern: Eine Zahl mit Komma ist also durch zwei ganze Zahlen dargestellt. Dabei bedeuten negative Exponenten eine Kommaverschiebung nach links, positive eine nach rechts. Man kann jetzt auch Zahlen darstellen, die wesentlich größer sind, als der normale Integerbereich es zulassen würde. Dafür benutzt man die Kommaverschiebung nach rechts. Die Zahl 1 000 000 000 (1 Milliarde) etwa läßt sich durch eine Mantisse von 1 und einen Exponenten von 9 darstellen. Man schreibt 1e9, wobei das „e“ als Trennung zwischen Mantisse und Exponent dient. In dieser Form, die schon im Beispiel am Anfang des Artikels verwendet wurde, versteht auch ein Computer die Eingabe dieser Zahl.
Bei unserem DM-Betrag setzten wir still-

schweigend voraus, daß das Komma beim Exponenten = 0 hinter der letzten Ziffer der Mantisse (also am rechten Ende) stehen würde. In der Praxis verwendet man allerdings die sogenannte „normalisierte Darstellung". Dabei steht das Komma hinter der ersten Ziffer, die ungleich Null ist. Diese Darstellung ist auch als wissenschaftliche Darstellung bekannt und wird von allen gängigen Taschenrechnern unterstützt. Unser DM-Betrag von „23,45" wird dann repräsentiert durch die Mantisse „2,345" und den Exponenten „1". Eine normalisierte Dezimalzahl beginnt also stets mit einer Ziffer zwischen „1" und „9", gefolgt von einem Komma und eventuell weiteren Ziffern. Eine normalisierte Binärzahl hingegen beginnt stets mit einer „1", da ja im Binärsystem nur zwei Ziffern existieren, nämlich „0" und „1". Dieser ersten Eins folgen dann das Komma sowie weitere Ziffern. Die Mantisse kann nun in zwei Teile zerlegt werden: Den Vorkommateil (also die führende „1") und den Nachkommateil. Letzteren nennt man auch „fraction", also Bruchteil. Die Nachkommastellen bezeichnet man daher auch als Binärbruch. *Tabelle 1* gibt weitere Erläuterungen zur Exponentialdarstellung.

Bei der Abspeicherung von Fließkommazahlen im Computer verwendet man in der Regel Vielfache von Bytes, z. B. 2 Byte (16 Bit) oder 4 Byte (32 Bit) oder auch noch mehr. Das Rechnen mit überlangen Integerzahlen wird von fast jedem Mikroprozessor unterstützt. Mit Hilfe eines Übertrags-Bits (Carry) kann man die Rechenoperationen einer CPU auf nahezu beliebig lange Integerzahlen ausdehnen. Wählt man zur Darstellung einer Zahl für die Mantisse n Bits, so lassen sich 2^n verschiedene Mantissen darstellen. Verwendet man für den Exponenten m Bits, so gibt es 2^m verschiedene Exponenten. Es lassen sich also $2^n \times 2^m$ verschiedene Gleitkommazahlen im Computer darstellen (nicht berücksichtigt, daß durch die normalisierte Darstellung eine weitere Einschränkung vorgenommen wird). In der Praxis eingesetzte Werte sind zum Beispiel n = 24 und m = 7. Damit kann man einen Zahlenbereich von 1,2e−38 bis 3,4e+38 abdecken. Das klingt eigentlich ganz gut ...

Ein Beispiel

Betrachten wir eine Fließkommadarstellung, bei der für die Mantisse 4 Bit und für den Exponenten 2 Bit verwendet werden. Bei normalisierter Darstellung können damit 32 verschiedene Zahlenwerte dargestellt werden, 33 wenn man die Null mit dazu nimmt. Das erste Bit der Mantisse

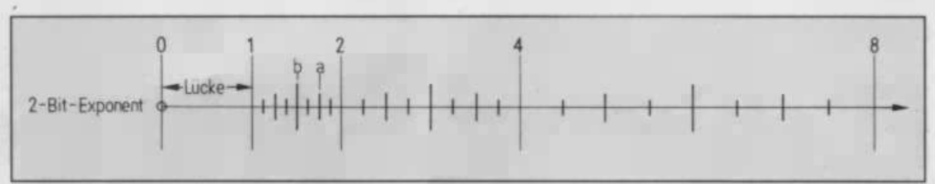


Bild 1. Der Zahlenstrahl bei verschiedenen Darstellungen

muß ja immer eine „1" sein, daher erhalten wir nur acht verschiedene Mantissenwerte statt der erwarteten 16 ($16 = 2^4$). In *Bild 1* wurden die ersten 24 normalisierten Fließkommazahlen sowie die Null auf einem Zahlenstrahl eingetragen. Bei der Betrachtung von Bild 1 fallen mehrere Dinge auf:

1. Mit wachsendem Exponentem nimmt der (absolute) Abstand zwischen den einzelnen Zahlenwerten zu.
2. Zwischen dem Wert 0 und der ersten Zahl ungleich Null besteht eine verhältnismäßig große Lücke.

Diese Lücke bereitet vor allem bei Divisionen Schwierigkeiten. So gibt zum Beispiel für die im Bild 1 markierten Werte a und b folgende Befehlssequenz trotz Sicherheitsabfrage den Fehler „Division durch Null":

if a <> b then x := 1 / (a – b)

Der Grund: a ist offensichtlich ungleich b. Mathematisch erwartet man also, daß die Differenz von a und b ungleich Null ist, die anschließende Division im „then"-Zweig also erlaubt ist. Bei der Berechnung der Differenz von a und b jedoch fällt das Ergebnis in die Lücke vor der Null und das berechnete numerische Resultat ist Null. Folgende Programmsequenz würde den Fehler vermeiden:

if (a – b) <> 0 then x := 1 / (a – b)

Hier wird schon bei der Abfrage im „if"-Zweig die echte Differenz gebildet und der „then"-Zweig nicht mehr durchlaufen.

Die Maschinenzahlen

Wir nennen die Zahlen, die der Computer exakt darstellen kann, im Unterschied zu den reellen Zahlen „Maschinenzahlen". Man kann die oben begonnene Liste der Auffälligkeiten von Bild 1 fortsetzen:

3. Es gibt keine beliebig großen, beliebig kleinen oder beliebig dicht nebeneinander liegenden Maschinenzahlen.
4. Werden zwei Maschinenzahlen miteinander verknüpft (addiert, subtrahiert, ...), so ist das Ergebnis im allgemeinen keine Maschinenzahl (siehe obiges Beispiel).

Es treten also an mehreren Stellen Ungenauigkeiten auf:

- Bei der Eingabe der Daten, da von der unendlichen Menge der reellen Zahlen auf die endliche Menge der Maschinenzahlen abgebildet werden muß.
- Beim Rechnen mit den Daten, da die Ergebnisse der einzelnen Operationen selbst meist keine Maschinenzahlen sind und daher nur mit einem gewissen Fehler abgespeichert werden können.
- Die sogenannten Verfahrensfehler treten vor allem bei komplizierteren mathematischen Funktionen auf, z. B. Iterationen zur Berechnung von Sinuswerten oder Exponentialfunktionen.

Die Ursachen dieser Fehler und ihre Fortpflanzung bei der weiteren Verarbeitung sind ein wichtiger Aspekt der Numerik. Man sagt von einem Algorithmus, daß er „numerisch stabil" ist, wenn man für die Rechenfehler stets eine obere Schranke angeben kann, oder, noch besser, wenn sich (speziell bei Iterationen) die Rechenfehler gegenseitig kompensieren. Viele mathematische Formeln sind aus numerischen Gründen ungeeignet, auf dem Computer berechnet zu werden. Oft aber genügen schon geringe Umstellungen bzw. Modifikationen, um den Algorithmus numerisch zu stabilisieren.

Periodische und unendliche Brüche

Ein weiteres Problem ist die Darstellung unendlich langer Zahlen: Die Zahl ⅓ ist ausgeschrieben 0,33333333... Werden nur endlich viele Stellen gespeichert – sagen wir 6 – so entspricht dies der Zahl 0,333333. Die Multiplikation mit drei liefert das Ergebnis 0,999999 – während das exakte Ergebnis natürlich Eins ist. Hier hat der Mensch eindeutig Vorteile, indem er Brüche wie ⅓ oder Ausdrücke wie $\sqrt{2}$, π, e^4 usw. nicht auswertet, sondern symbolisch verarbeitet. Um $\sin(\pi)$ auszurechnen braucht niemand einen Taschenrechner, man weiß eben, das ergibt Null. Der Computer jedoch kann niemals exakte Ergebnisse liefern, da er schon die Zahl π nicht genau darstellen kann. Und weil Computer nicht im Dezimal- sondern im Binärsystem arbeiten, tauchen für den Laien manchmal unerwartete Fehler auf: Bereits den einfachen Bruch ⅒ kann der Computer nicht exakt darstellen, da sich dafür ein unendlicher Binärbruch ergibt. Zweierpotenzen hingegen kann er sehr gut darstellen (also



Kompakt und modular:
Prozeßsteuerungen von kws.
Netzwerkkarte
auch für Glasfaseranschluß. Token-Passing-Struktur, speziell für Echtzeit-Applikationen.
Grafikkarte
640 x 400, 16 Farben, Look-up-table
Floppy + SCSI
intelligentes Interface mit 68070 CPU
SYSCON
System-I/O, Watchdog, Not-Aus, RAM, EPROM
CPU 68000
dual ported RAM, EPROM, V.24, I/O-Businterface
Befehlstastatur
kundenspezifisch, max. 48 Tasten, LC-Display 40 x 2, beleuchtet
AD-8
8 x 13 Bit Analogeingang, integrierend, Kanalanzeige
Mit VME-Bus...
als leistungsfähigem Standardbus
DI-16
Digitaleingang 2 x 8 Bit, Einzelbit-Interrupts. 24 V.
DO-16
Digitalausga 2 x 8 Bit, 1 A pro Kanal kurzschlußfest, Anzeigen, Ausgangsüberwachung. 24 V.
VME-CPU 01, mit 256 KB RAM und 256 KB EPROM
DA-4
4 x 12 Bit Analogausgan 0...10 V, – 10...10 V, 0...20 mA, 4...20 mA
...und I/O-Bus
mit gekapselten Module konsequent opto-entkoppe und störsich
Modul DO-16, geöffn
kws
Steiner + Raase
kws Computersysteme GmbH
Dr. Kurtze und Partner
Einsteinstraße 22
D-7505 Ettlingen
Telefon (07243) 78068
Meßsysteme und Subsysteme
Stand-alone und über Glasfasernetz
Leistungsstark, flexibel, universell und preisgünstig
Rufen Sie uns an!

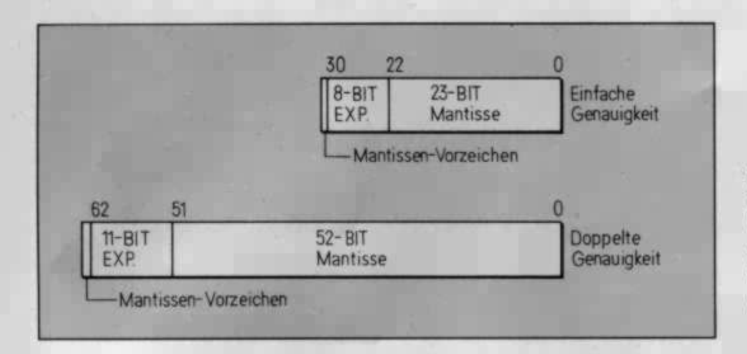


Bild 2. Format von Zahlen mit einfacher (single-) und doppelter Genauigkeit (double-precision)

1; 2; 128; 2048 sowie 0,5; 0,25; 0,0625 usw.)
Nach soviel Ungenauigkeiten taucht vielleicht die Frage auf, ob man denn eigentlich überhaupt mit Computern rechnen kann – doch, man kann. Um die auftretenden Fehler möglichst gering zu halten und das Verhalten der arithmetischen Operationen für alle Computer zu standardisieren, wurde vom IEEE (Institute of Electrical and Electronical Engineers) die Arbeitsgruppe „P754" ins Leben gerufen. Diese hatte zur Aufgabe, einen für alle Computerhersteller und Anwender akzeptablen Vorschlag für die Darstellung, Verarbeitung und Fehlerbehandlung von Fließkommazahlen auszuarbeiten, so daß Kompatibilität zwischen allen Computersystemen und Programmiersprachen erreicht werden kann. Die ganze Sache ist übrigens kein alter Hut: erst 1981/1982 wurde die Fließkommadarstellung zur Normung vorgeschlagen. Mittlerweile hat sich der Standard schon stark verbreitet, so erfüllen ihn z. B. moderne Fließkommaprozessoren wie der 80287/80387 von Intel und 68881/68882 von Motorola in allen Punkten.

IEEE P754

Bereits in mc 4/88 wurden einige Hinweise zu den in der Norm getroffenen Festlegungen für die Handhabung von Fließkommazahlen gegeben. Im Folgenden werden alle Punkte des Vorschlags abgehandelt.
Gerechnet wird im Binärsystem. Das gilt sowohl für die Mantisse als auch für den Exponenten. Die Verwendung des Binärsystems ist nicht unbedingt zwingend, man könnte auch mit Dezimal- oder Hexadezimalzahlen arbeiten, jedoch wird das Binärsystem von den Mikroprozessoren am effizientesten unterstützt. Wie schon erwähnt, legt die IEEE-Spezifikation auch das genaue Format von Fließkommazahlen fest. Zwei Formate wurden vorgesehen:
☐ Single-Precision (einfache Genauigkeit)
☐ Double-Precision (doppelte Genauigkeit)
Bild 2 veranschaulicht die beiden Formate. Einfache Genauigkeit arbeitet mit 32 Bit, also 4 Bytes, doppelte Genauigkeit hingegen mit 64 Bit, also mit 8 Bytes. Ein Teil davon entfällt jeweils auf den Exponenten, ein Vorzeichenbit sowie die Mantisse. Die Mantisse wird stets als positive Zahl gespeichert, für das Vorzeichen wird ein eigenes Bit verwendet. Die Mantissen werden also nicht wie Integerzahlen im Zweierkomplement gespeichert, sondern getrennt in Betrag und Vorzeichen. Der Exponent wiederum wird stets mit einem „Bias" verknüpft. Dieser Bias ist eine Konstante, die vor der Abspeicherung zum Wert des Exponenten addiert wird. Dadurch erreicht man, daß das Exponentenfeld einer Fließkommazahl stets eine positive Zahl ist. Für die betragsmäßig kleinsten Zahlen steht im Exponentenfeld der Wert Null, was die Handhabung des Unterlaufes vereinfacht (eine Fließkommazahl ist dann Null, wenn ihr Exponent und ihre Mantisse Null sind).
Durch die Wahl von Vielfachen von Bytes können die Fließkommazahlen effizient gespeichert und übertragen werden. Moderne 32-Bit-Prozessoren können eine Zahl einfacher Genauigkeit sogar in CPU-Registern verarbeiten. Das 32-Bit-Format ist ein Kompromiß zwischen Genauigkeit und Rechengeschwindigkeit zugunsten der Rechengeschwindigkeit, während das 64-Bit-Format vor allem wegen hoher Genauigkeit gewählt wurde. Beim IEEE-Format werden die Mantissen stets in normalisierter Form abgespeichert. Jede Zahl hat das in *Tabelle 2* angegebene Format mit einer führenden Eins. Dadurch wird der zur Verfügung stehende Platz optimal genutzt, weil Vornullen, die ja eigentlich überflüssig sind, nicht mit abgespeichert werden. Und da ja alle Zahlen gleich anfangen ist es auch unnötig, die führende 1 mit abzuspeichern, man merkt sie sich nur in Gedanken. Dadurch erhöht sich die Mantissen-Länge um ein Bit. Man spricht auch von einer „impliziten" Eins. Von einer nicht-normalisierten Zahl gelangt man zur normalisierten Zahl, indem man das Komma verschiebt, bis vor dem Komma noch genau eine 1 steht. Bei jeder Komma-Verschiebung wird der Exponent entsprechend der Verschiebung erhöht bzw. erniedrigt. Als drittes Format fordert IEEE ein „extended-precision"-Format (erhöhte Genauigkeit). Dieses Format soll stets zum Rechnen und für Zwischenergebnisse verwendet werden. Es muß über eine größere Genauigkeit als „Double-Precision" verfügen, soll aber gegenüber diesem Format nicht mehr als den doppelten Speicherplatz benötigen. So verwendet Motorola in seinem Fließkommaprozessor 68881/68882 ein Format von 80 Bit, 64 Bit für die Mantisse und 16 Bit für den Exponenten. Durch die Verwendung des „extended"-Formats sollen speziell bei Iterationen die Rechenfehler weiter abgesenkt werden, da Genauigkeitsverluste normalerweise nur in den letzten Bits auftreten, die später bei der Darstellung in einfacher oder doppelter Genauigkeit ohnehin nicht abgespeichert werden. Für den Anwender soll das Format „extended precision" nicht zugänglich sein; er darf lediglich die beiden Formate „single" und „double precision" sehen. Für das genaue Format der erhöhten Genauigkeit wird keine Festlegung getroffen, hier hat der Hersteller bzw. Programmierer die freie Wahl. Die IEEE-Vorschläge sind so gestaltet, daß eine Implementation wahlweise nur in Software, nur in Hardware oder auch in Mischformen erfolgen kann.

Rundung

Das IEEE fordert für alle arithmetische Funktionen eine so hohe Rechengenauigkeit, daß das ermittelte Ergebnis um höchstens die halbe Wertigkeit des letzten Bits der Mantisse vom korrekten Ergebnis abweicht. Um das zu erreichen, müssen Zwischenergebnisse mit mindestens 1 Bit mehr berechnet werden, als für die Darstellung von Zahlen einfacher bzw. doppelter Genauigkeit benötigt wird. IEEE fordert sogar

Tabelle 2: Normalisierte Zahlen

Zahl vor der Normalisierung: Normalisierte Zahl:	0.00100100101 1.00100101000	Exponent: 0 Exponent: –3
Verschiebung des Punktes um 3 Stellen nach rechts → Exponent: = Exponent – 3		
Zahl vor der Normalisierung: Normalisierte Zahl:	100.100000000 1.00100000000	Exponent: 3 Exponent: 5
Verschiebung des Punktes um zwei Stellen nach links → Exponent: = Exponent + 2		

3 Bit, deren Bedeutung *Bild 3* zeigt. Das Guard-Bit (Wächter-Bit) wird berechnet, um eine eventuell nötige Normalisierung der Zahl um ein Bit nach links zu ermöglichen. Das Round-Bit wird zum Auf- bzw. Abrunden benötigt. Das Sticky-Bit schließlich ist ein logisches Oder von allen Bits, die im internen Format rechts hinter dem Round-Bit liegen. Das Sticky-Bit ist wichtig beim Runden für Intervallarithmetik. Im einzelnen sollen in einer Implementierung nach IEEE die folgenden Rundungsarten und -genauigkeiten enthalten sein.

a) default rounding mode (round to nearest): Zur nächsten Zahl auf- bzw. abrunden. Der dabei entstehende Fehler ist ±½ LSB (LSB = „least significant bit", also das Bit mit der niedrigsten Wertigkeit in der Mantisse).
b) directed-rounding-mode: Vom Anwender bestimmbar wird wahlweise immer aufgerundet (zur nächstgrößeren Zahl) oder immer abgerundet (zur nächstkleineren Zahl). Der hierbei auftretende Fehler liegt bei maximal ±1 LSB.

Die Rundungsgenauigkeit soll immer dem jeweiligen Format entsprechen, also 32 oder 64 Bit. Zusätzlich soll die Möglichkeit bestehen, eine 64-Bit-Zahl auf 32-Bit zu runden, um sie später in einfacher Genauigkeit weiter zu verarbeiten.

Operationen

Die IEEE-Spezifikation legt nur die grundsätzlichen arithmetischen Operationen fest: Addition, Subtraktion, Multiplikation, Division, Ziehen der Quadratwurzel, Rest- und Vergleichsoperationen. Höhere mathematische Funktionen (wie z. B. sin, cos, e-Funktion, log) bauen auf diesen Basisfunktionen auf. Sie müssen ohnehin durch iterative Verfahren berechnet werden, im Gegensatz zu den Basisfunktionen, die gewissermaßen geradeaus („straight-forward") programmiert werden. Alle arithmetischen Basisfunktionen müssen den angegebenen Regeln der Rundung gehorchen. Die Quadratwurzel wurde hier eigentlich nur erwähnt, um eine Ausnahme festzulegen:

$$\sqrt{-0} = -0$$

Normalerweise ist die Wurzel aus einer negativen Zahl undefiniert. Da jedoch in der IEEE-Spezifikation die Möglichket besteht, auch der Null ein Vorzeichen zu geben, gilt diese zusätzliche Regel. Die negative Null spielt vor allem beim affinen Abschluß der Zahlen eine Rolle. Beim projektiven Abschluß darf die negative Null nicht erzeugt werden. Auch das muß von der Rundungs-Software berücksichtigt wer-

Tabelle 3: Anforderungen an Zahlenbereich und Stringlänge für Zahlenkonvertierungsalgorithmen in Arithmetikpaketen

Format	Dezimal → (max. Wert)	Binär (max. Ziffern)	Binär → (max. Wert)	Dezimal (max. Ziffern)
Diese Formate müssen bei der Eingabe akzeptiert werden:				
Single	10e9 −1	99	10e9 −1	54
Double	10e19 −1	999	10e17 −1	341
In diesem Bereich muß korrekt gerundet werden:				
Format	Dezimal → (max. Wert)	Binär (max. Ziffern)	Binär → (max. Wert)	Dezimal (max. Ziffern)
Single	10e9 −1	13	10e9 −1	13
Double	10e19 −1	27	10e17 −1	27

Tabelle 4: Einige Additionen

32-Bit-Format	*24-Bit-Exponent*	*24-Bit-Mantisse*	*Kommentar*
3F80 0000 + 3F80 0000	2^0 + 2^0	× 1,00...00 × 1,00...00	Der Übertrag im Ergebnis macht eine Verschiebung nach rechts erforderlich. Das Ergebnis ist exakt, Rundung nicht notwendig.
 = 4000 0000	2^0 2^1	× 10,00...00 × 1,00...00	
3F80 0000 + 3400 0000	2^0 + 2^{-23}	× 1,00...00 × 1,00...00	2^{-23} wird als Eins auf der letzten Stelle addiert. Das Ergebnis ist exakt, Rundung nicht notwendig.
= 3F80 0001	2^0	× 1,00...01	
3F80 0000 + 3380 0000	2^0 + 2^{-24}	× 1,00...00 × 1,00...00	2^{-24} wird als ½ LSB addiert, bei Rundung nach oben oder zum Nächsten ergibt sich hier eine 1, bei Rundung nach unten eine 0.
 = 3F80 0000 3F80 0001	2^0 2^0 2^0	× 1,00...00:1 × 1,00...00 × 1,00...01	oder
3F80 0000 + 0080 0000	2^0 + 2^{-126}	× 1,00...00 × 1,00...00	2^{-126} liegt weit rechts vom LSB, daher wird das „Sticky Bit" gesetzt, nur Rundung nach oben setzt LSB auf 1.
 = 3F80 0000 3F80 0001	2^2 2^0 2^0	× 1,00...001:...01 × 1,00...00 × 1,00...01	oder

Hexadezimal-Darstellung		Binärdarstellung seee	eeee	efff	ffff	ffff	ffff	ffff	ffff
3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
+ 3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
= 4000	0000	0100	0000	0000	0000	0000	0000	0000	0000
3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
+ 3400	0000	0011	0100	0000	0000	0000	0000	0000	0000
= 3F80	0001	0011	1111	1000	0000	0000	0000	0000	0000
3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
+ 3380	0000	0011	0011	1000	0000	0000	0000	0000	0000
= 3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
3F80	0001	0011	1111	1000	0000	0000	0000	0000	0001
3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
+ 0080	0000	0000	0000	1000	0000	0000	0000	0000	0000
= 3F80	0000	0011	1111	1000	0000	0000	0000	0000	0000
3F80	0001	0011	1111	1000	0000	0000	0000	0000	0001

den. Der Unterschied zwischen projektivem und affinem Abschluß der reellen Zahlen wurde bereits in mc 4/88 (S. 49) gut erklärt.

Format-Konvertierungen

Vom Fließkommapaket müssen folgende Konvertierungen unterstützt werden:

a) single precision → double precision (einfache → doppelte Genauigkeit)
b) double precision → single precision (doppelte → einfache Genauigkeit)
c) integer → floating point (ganzzahlig → Gleitkomma)
d) floating point → integer (Gleitkomma → ganzzahlig)
e) decimal string → floating point (Zeichenkette → Gleitkomma)
f) floating point → decimal string (Gleitkomma → Zeichenkette)

Gerade die letzte Operation gestaltet sich bei der Realisierung extrem aufwendig. Zusätzlich werden für die Funktionen (e) und (f) die Mindestanforderungen an Zahlenbereiche und String-Längen festgelegt, die bei der Umwandlung verarbeitet werden sollen (*Tabelle 3*).

Betrachten wir nun zur Erholung von soviel Theorie einige Beispiele (*Tabelle 4*): Hier werden jeweils zwei Zahlen addiert. Anhand der Kommentare lassen sich die einzelnen Schritte bei der Fließkomma-Addition nachvollziehen.

NaN's und unendlich...

Werden zwei sehr große positive Zahlen addiert, so kann es zum Überlauf kommen. Für diesen Fall sieht die IEEE-Spezifikation die Darstellung des Wertes „+∞" (positiv unendlich) vor, analog gibt es den Wert „−∞". Auch bei der Division durch Null wird +∞ bzw. −∞ erzeugt. Die Abkürzung NaN steht für „not a number", also „keine Zahl". Werden Operationen mit unzulässigen Argumenten aufgerufen (etwa eine Wurzel mit negativem Argument), dann erzeugen sie als Ergebnis eine NaN. Dabei werden zwei NaNs unterschieden: „signalling" (= trapping) und „nonsignalling" (= nontrapping) NaN. Erstere löst einen Interrupt (sogenannte Exception) aus, während die zweite das nicht tut. Man nennt die zweite NaN auch „quiet NaN". Welche der beiden NaNs jeweils erzeugt wird, kann der Anwender über die globale Betriebsart des Fließkommapaketes entscheiden. Bei der Verwendung von nonsignalling-NaNs müssen an geeigneten Stellen Abfragen durchgeführt werden, die das Auftreten von NaNs erkennen. Dazu gibt es die Abfragebedingung „unordered" (nicht vergleichbar). Signalling NaNs hingegen lösen bei ihrer Erzeugung einen Interrupt aus. In einer entsprechenden Interruptbehandlungsroutine kann dieser vom Anwendungsprogramm abgefangen und ausgewertet werden. So ergibt sich beispielsweise die Möglichkeit, eine Wurzel aus negativen Zahlen (zum Rechnen mit komplexen Zahlen) in der Interruptroutine zu emulieren.

SIGNIFICANT BITS
LEAST SIGNIFICANT BIT
GUARD BIT
ROUND BIT
STICKY BIT

Bild 3. Zusätzliche Bits für Rundungszwecke

Tabelle 5: Gleitkomma-Format bei einfacher und doppelter Genauigkeit

Speicherformate Einfache Genauigkeit: s eeee eeee fff ffff ffff ffff ffff ffff Doppelte Genauigkeit: s eee eeee eeee ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff		
Beschreibung	*einfache Genauigkeit*	*doppelte Genauigkeit*
Feldgrößen (in Bit) s = Vorzeichen e = Exponent mit Bias f = Mant.-Nachkommast. Gesamt	 1 (Bit 31) 8 (Bit (30...23) 23 (Bit 22...0) 32	 1 (Bit 63) 11 (Bit 62...52) 52 Bit 51...0) 64
Vorzeichen Positive Mantisse, s = Negative Mantisse, s =	 0 1	 0 1
Normalisierte Zahlen Bias von e (Exponent) Wertebereich von e Wertebereich von f Mantisse komplett Zahlen-Repräsentation	 +127 ($7F) 0 < e < 255 ($FF) Null oder Nicht-Null 1,f $(-1)^s \times 2^{(e-127)} \times 1,f$	 +1023 ($3FF) 0 < e < 2047 ($7FF) Null oder Nicht-Null 1,f $(-1)^s \times 2^{-e-1023}) \times 1,f$
Denormalisierte Zahlen e = Format-Minimum = Bias von e Wertebereich von f Mantisse Zahlen-Repräsentation	 0 ($00) +126 ($7E) Nicht-Null 0,f $(-1)^s \times 2^{(-126)} \times 0,f$	 0 ($000) +1022 ($3FE) Nicht-Null 0,f $(-1)^s \times 2^{(-1022)} \times 0,f$
Nullen (vorzeichenbehaftet) e = Format-Minimum = f = Mantisse-Nachkomma =	 0 ($00) 0,f = 0,0	 0 ($000) 0,f = 0,0
Unendlich (vorzeichenbehaftet) e = Format-Maximum = f = Mantisse =	 255 ($FF) 0,f = 0,0	 2047 ($7FF) 0,f = 0,0
NANs (Not-A-Number) s = e = Format-Maximum = f =	 nicht signifikant 255 ($FF) Nicht-Null, und zwar: ,1xxxx...xxxx Non-Signalling ,0xxxx...xxxx Signalling mit xxxx...xxxx Nicht-Null	 nicht signifikant 2047 ($7FF) Nicht-Null, und zwar: ,1xxxx...xxxx Non-Signalling ,0xxxx...xxxx Signalling mit xxxx...xxxx Nicht-Null
Wertebereiche (gerundet) größte pos. norm. Zahl kleinste pos. norm. Zahl kl. pos. denorm. Zahl	 $3,4 \times 10^{38}$ $1,2 \times 10^{-38}$ $1,4 \times 10^{-45}$	 18×10^{307} $2,2 \times 10^{-308}$ $4,9 \times 10^{-324}$

Um die Werte +∞ und −∞ sowie die NaNs darzustellen, hat man den Zahlenbereich geringfügig eingeschränkt, indem der Exponent $FF bzw. $7FF als Kennzeichnung für diese besonderen Zahlen dient.

Nicht-normalisierte Zahlen

Bei der Diskussion des Zahlenstrahles in Bild 1 fiel die unschöne Lücke im Bereich der Null auf. Der sogenannte „flush-to-zero"-Effekt, der auftritt, wenn Ergebnisse zu nahe an der Null liegen, kann fatale Folgen haben (z. B. bei anschließender Division). Die IEEE-Spezifikation sieht aus diesem Grund neben den normalisierten Zahlen, bei denen vor dem Komma immer eine 1 steht, auch denormalisierte (oder nicht-normalisierte) Zahlen vor. Bei diesen steht vor dem Komma statt einer Eins eine Null. Zur Kennzeichnung, daß eine Zahl denormalisiert ist, dient wiederum das Exponentenfeld: Der Wert $00 wird benutzt, um denormalisierte Zahlen zu kennzeichnen. *Tabelle 5* schließlich gibt einen Überblick über die Zahlenbereiche, Darstellungsformate und den Aufbau der Fließkommazahlen mit einfacher und doppelter Genauigkeit. Durch die Einführung der denormalisierten Zahlen werden bei einfacher Genauigkeit nochmals 23 Binärstellen gewonnen, bei doppelter 52 Stellen. Natürlich sind die denormalisierten Zahlen mit größeren Fehlern behaftet als die normalisierten, da die signifikanten (aussagekräftigen) Bits in der Mantisse relativ weit hinten stehen. Man spricht jetzt von graduellem Unterlauf, wenn das Ergebnis einer Operation eine denormalisierte Zahl ist. Endgültiger Unterlauf tritt erst dann auf, wenn die gesamte Mantisse Null ist. *Bild 4* zeigt im Vergleich zu Bild 1 den Gewinn an zusätzlichen Zahlen in der Nähe der Null.

Ausnahmen (Exceptions)

Exceptions sind Interrupts, die von der Fließkomma-Einheit ausgelöst werden, sei diese Einheit ein Softwarepaket oder ein eigener Prozessor. Die Interrupts signalisieren jeweils Fehler einer bestimmten Klasse. Jede Klasse verfügt über eine eigene Interruptmaske und kann wahlweise freigegeben oder abgeschaltet werden. Es werden 5 Klassen unterschieden:

1. Invalid Operation: Unzulässiger Befehl oder unzulässiges Rechenergebnis.
2. Division by Zero: Division durch Null
3. Overflow: Überlauf
4. Underflow: Unterlauf
5. Inexact: Ungenaues Rechenergebnis

Letzteres tritt auf, wenn beim Runden Bits abgeschnitten wurden.

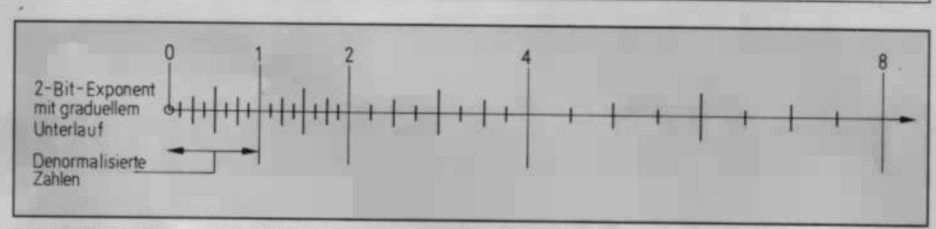


Bild 4. Erweiterung der Fließkommazahlen mit denomalisierten Zahlen

Tabelle 6: Lexikon

Begriff	*Übersetzung*	*Bedeutung*
Bias	Abgleich	Eine Konstante, die zum Wert des Exponenten addiert wird. Dadurch wird erreicht, daß der abgespeicherte Exponent stets eine positive Zahl ist. Der Wert des Exponenten wird minimal bei Null.
binary floating point number	binäre Fließkommazahl	Ein Bitstring, der durch drei Komponenten definiert wird: ein Vorzeichen, einen vorzeichenbehafteten Exponenten und eine Mantisse. Der numerische Wert – sofern er definiert ist – ergibt aus dem Produkt der Mantisse mit dem Exponenten potenziert zur Basis zwei.
Carry	Übertrag	Das Carry-Bit wird im Mikroprozessor benutzt, um den Übertrag auf die nächst höhere Stelle anzuzeigen.
Exception/Trap	Ausnahme/Falle	Eine Ausnahme tritt unter bestimmten Bedingungen auf, wie etwa der Division durch Null, oder dem Versuch, die Wurzel aus einer negativen Zahl zu berechnen. In einem Exception-handler oder Trap-handler wird die Ausnahme dann „behandelt", also z. B. eine Fehlermeldung ausgegeben. Eine Exception verhält sich asynchron wie ein Interrupt.
exponent	Exponent	Diese Komponente einer Fließkommazahl gibt die Zweierpotenz an, mit der die Mantisse multipliziert werden muß, um den Wert einer Fließkommazahl zu erhalten.
Expression	Ausdruck	Unter einem Ausdruck versteht man meistens einen arithmetischen Term, eine einfache Formel. Beispiel: (a + b) × c oder 4/3.14 oder 5
Mantissa	Mantisse	Diese Komponente einer Fließkommazahl besteht aus einem impliziten oder expliziten Bit links vom Komma und einem Binärbruch rechts vom Komma.
Integer	Ganze Zahl	Integer-Werte sind stets ganzzahlig. Zur Darstellung von gebrochenen und reellen Zahlen werden die Fließkommazahlen verwendet.
Fraction	Bruchteil	Der Teil der Mantisse, der rechts vom Komma liegt.
NaN = Not-A-Number	„keine Zahl"	Spezielle Pseudozahlen, die zur Anzeige von besonderen Fehlersituationen dienen.
Non-Signalling (non-trapping, quiet)	Nicht-signalisierend	Es werden zwei Arten von NaNs unterschieden. Die nicht-signalisierenden führen zu keiner Exception. Sie werden daher auch als „quiet NaNs" (= ruhige NaN) bezeichnet.
Signalling (trapping)	Signalisierend	Die signalisierenden NaNs führen nach ihrer Erzeugung zu einer Exception, d. h. das Programm wird an dieser Stelle unterbrochen bzw. abgebrochen.

Eine Erläuterung der genauen 5Rahmenbedingungen und Sonderfälle für Erzeugung und Behandlung dieser Interrupts würde hier zu weit führen. Von Interesse sind zunächst nur die verschiedenen Klassen und die Tatsache, daß jede Klasse einzeln ein- und ausgeschaltet werden kann. Bei abgeschalteten Klassen wird mit den Werten +∞, −∞ sowie den NaNs weitergerechnet. Die IEEE-Spezifikation sieht daher auch für alle Basisfunktionen entsprechende Algorithmen für die Verknüpfung dieser speziellen Operanden vor (wie die Summe von +∞ und −∞, 3.456 + NaN usw.).

Fortsetzung in Ausgabe 11/88. Sie erhalten die nächste Ausgabe am 24. Oktober.

Ulrich Gärtner

Der PC als Steuerungsrechner

Teil 1: Die mc-E/A-Karte

Der PC läßt sich mit den entsprechenden Erweiterungskarten als Steuerungsrechner für vielfältige Aufgaben einsetzen. Die mc-E/A-Karte ist die Schnittstelle zwischen dem PC, der Außenwelt und speziellen Karten für die Steuerungstechnik, die in der nächsten Ausgabe beschrieben werden.

Damit ein Rechner mit der Außenwelt kommunizieren kann, benötigt er eine Schnittstelle, die sich nach der Art des zu verarbeitenden Signals richtet. Sollen kontinuierliche Signale erfaßt werden, benötigt man einen A/D-Umsetzer. Bestehen die Signale nur aus zwei Zuständen, high (H) oder low (L), kann die mc-E/A-Karte verwendet werden.

Signale mit nur zwei Zuständen kommen recht häufig vor. Will man den Zustand eines Schalters (z. B. DIP-Schalter) oder eines Ventils (offen oder geschlossen) abfragen, so treten nur die Zustände offen („0") oder geschlossen („1") auf. Ebenso kann es auch erwünscht sein, daß der Rechner externe Einheiten steuert, wobei man wieder zwischen den Signalarten unterscheiden muß. Sollen kontinuierliche Signale ausgegeben werden, muß man auf einen D/A-Wandler zurückgreifen. Sollen allerdings Relais, Schalter oder Ventile gesteuert werden, reichen wieder nur zwei Zustände (offen oder geschlossen) aus. Mit der mc-E/A-Karte, auch IO32/32 genannt, sprechen Anwendungen 32 Ausgabe- und 32 Eingabekanäle unter vier E/A-Adressen an, die man mit gewissen Einschränkungen frei wählen darf.

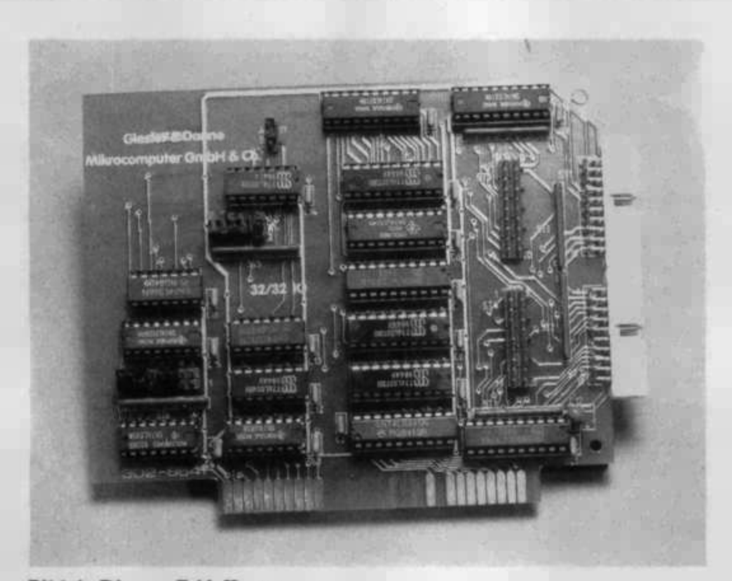


Bild 1. Die mc-E/A-Karte

Bei der Entwicklung der mc-E/A-Karte (*Bild 1*) wurde bewußt auf programmierbare Portbausteine verzichtet, um die Karte so transparent wie möglich zu lassen. Bei der Programmierung der Karte kann jeder Schritt leicht mit einem Oszilloskop nachverfolgt werden. Ebenso kann man sich durch kleine Testprogramme die Funktionsweise einzelner Einheiten der Karte mit dem Oszilloskop veranschaulichen. Besonders die Funktion des Decoders geht aus der Karte klar hervor.

Die Schaltung im Detail

Die Adreß-Decodierung aller PC-Erweiterungskarten ist ähnlich aufgebaut.

Aus dem Schaltplan (*Bild 2*) geht hervor, daß die Karte in drei Funktionsgruppen gegliedert ist:

- Dekodierteil (Schnittstelle zum PC-Bus),
- Eingabeeinheit
- Ausgabeeinheit

Die Adreßdekodierung führen die Bausteine IC3 und IC4 (7485, 4-Bit-Vergleicher) durch. Vier Adressen werden aus dem verfügbaren E/A-Adreßraum von 1024 Adressen dekodiert. Von den 1024 E/A-Adressen sind 512 für die Systemplatine (Adreß-Bit A9 = 0) und die restlichen 512 für Systemerweiterungen auf Karten (Adreß-Bit A9 = 1) reserviert. Die Basisadresse wird über A2...A9 auskodiert. IC3 vergleicht die Adreß-Bits A6...A9 mit der mit den Jumpern eingestellten Adresse. Sind sie gleich und gibt AEN (Adress Enable) den Gleichheitseingang frei, wird der untere Bereich (Adreß-Bits A2...A5) mit IC 4 decodiert. Bei der Wahl der Basisadresse ist zu beachten, wie der E/A-Adreßraum von IBM belegt wurde. Die E/A-Adressen 300H...31FH sind für Erweiterungskarten vorgesehen. Vor der Inbetriebnahme der Karte sollte man sich vergewissern, daß unter diesen Adressen nicht schon andere Erweiterungskarten , z. B. A/D- oder D/A-Wandler-Karte angesprochen werden.

Durch die Verknüpfung der Adreßdecodierung mit dem Adreß-Freigabe-Signal (Address Enable Signal) wird sichergestellt, daß die Karte nicht versehentlich während einem DMA-Zyklus auf den Bus zugreift. Ist die anliegende Adresse identisch mit der vom Jumper J1 festgelegten, wobei nach *Tabelle 1* der freie Adreßraum und die Basisadresse zu beachten sind, wird an IC 13 eine logische Eins ausgegeben. IC 13 erzeugt mit der Oder-Verknüpfung der Signale IORC (I/O Read Command) und IOWC (I/O Write Command) das Richtungsumschaltungs-Signal für den Bustreiber IC 2. Außerdem gewinnt es das Auswahlsignal für den 1-aus-4-Decoder IC 15.

Diese Decoder sind für die Auswahl der Eingänge bzw. der Ausgänge zuständig. Da der Pin 19 von IC 2 auf Masse liegt, ist der Datentreiber ständig aktiv. Bei einem Lesezugriff auf die Karte erscheint am DIR-Eingang von IC 2 (Datentreiber) ein Low, woraufhin der interne Datenbus (Datenbus der Karte) den Zustand des PC-Datenbusses bestimmt. Bei einem Schreibvorgang passiert im Prinzip das gleiche, nur das jetzt der PC-Datenbus den Zustand des internen Datenbusses bestimmt.

E/A-Baugruppe

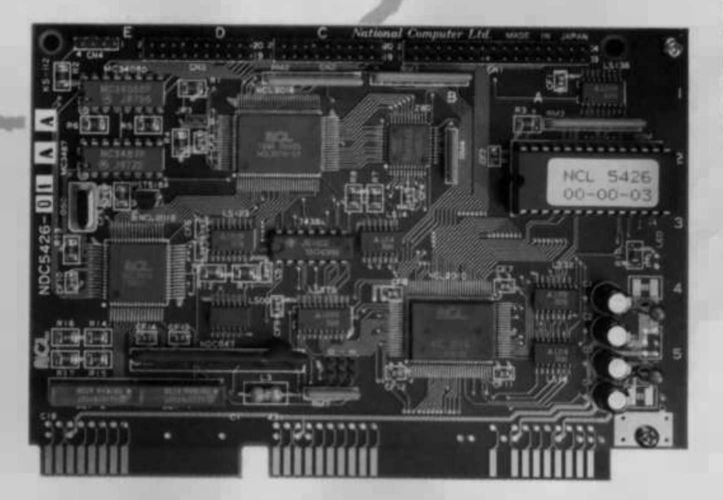
Die Schaltkreise IC 5, 6, 9, 11 (74LS273) dienen zur Ausgabe von Signalen. Zum Einlesen von Signalen mit TTL-Pegel werden die Schaltkreise IC 1, 7, 8, 10 (74LS374) verwendet.

Der Baustein 74LS273 ist ein 8faches flankengetriggertes D-Flipflop. Ein Flipflop ist eine Kippstufe, die im Gegensatz zum Schmitt-Trigger durch einen kurzen Impuls angestoßen wird, der die Schaltung in ei-



„0 1 0 1 0 0 1 1 1 0 1"

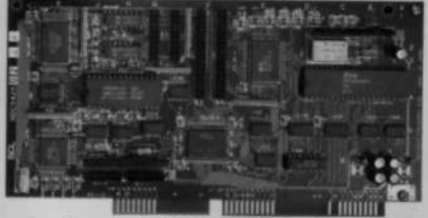
„0 1 0 1 0 0 1 1 1 0 1"



Der NCL-Controller steuert die Datenübertragung zwischen dem System und den Laufwerken. Mit einer CPU-Geschwindigkeit bis zu 25 MHz.

Der NCL-Controller läuft auf schnellsten Systemen - SYDEC liefert und berät auf schnellsten Wegen.

Der NCL-Controller paßt sich an die hohe Taktfrequenz von AT-Rechnersystemen an. Problemlos bis zu 25 MHz CPU-Geschwindigkeit.
Der hohe japanische Qualitätsstandard von NCL zeigt sich in der geringen Rücklaufquote von unter 0,5%. Und durch die um 1/3 kleinere Abmessung. Das Preis-Leistungs-Verhältnis von NCL-Controllern ist zur Zeit in der Computerindustrie unerreicht.
Zur Wahl stehen: Der NCL Harddisk-Controller und der NCL Combi-Controller für Harddisk und Floppy. Beide NCL-Controller passen in jedes IBM PC AT kompatible System.



Jetzt zu SYDEC: 1 - 100 Stück liefern wir prompt vom Lager. Innerhalb weniger Tage. Falls Sie ein fehlerhaftes Teil umtauschen wollen, haben Sie in spätestens 48 Stunden Ihr Ersatzteil. Und weil Geschwindigkeit nicht alles ist, bietet SYDEC einen Telefon-Sofortdienst für technische Fragen.

Was wollen Sie mehr? Ach ja, den NCL-Controller.

Fordern Sie Prospektmaterial über alle unsere Controller an. Anruf genügt: ☎ 0 80 62/50 17



Computerbauteile ohne Umwege

Autorisierter Fachhändler für: Microscience, NCL und Mitsubishi. SYDEC Deutschland GmbH. Technology Distributors.
Kirchdorfer Straße 14b · 8206 Bruckmühl

Tabelle 1: Einstellung der Steckbrücke J1

Position	Adreßleitungen
A	A9
B	A8
C	A7
D	A6
E	A5
F	A4
G	A3
H	A2

Tabelle 2: Die Logiktabelle des Bausteins 74LS273

Eingänge	Ausgänge			
	−MR	CP	Dn	Qn
Reset	0	X	X	L
Lade "1"	1	∧	h	H
Lade "0"	1	∧	l	L

h: Übernahme mit der Flanke von low nach high
l: Übernahme mit der Flanke von low nach high
X: Don't care
−MR: Master-Reset
CP: Takteingang (Clock Pulse)
∧: steigende Taktflanke

nen anderen Zustand bringt. Dieser wird bis zum nächsten Impuls gehalten. Ein flankengetriggertes Flipflop, allgemein D-Flipflop genannt, übernimmt die am Eingang anliegenden Informationen mit der steigenden Flanke am Takt-Eingang, somit also immer zu einem fest definierten Zeitpunkt. Der Baustein hat 8 Eingänge und 8 Ausgänge, dazu kommen die beiden Anschlüsse für die Versorgungsspannung und zwei Steuereingänge.
Die *Tabelle 2* zeigt die Logiktabelle für den 74LS273.

Bei einem Kaltstart wird das IC über die Reset-Leitung zurückgesetzt, so daß die Ausgänge low („0") liegen. Durch diese Maßnahme ist der Anfangszustand der Schaltung sichergestellt, den sie nach dem Einschalten der Anlage oder nach einem Stromausfall einnimmt. Die nachgeschaltete Elektronik kann an diesen Zustand angepaßt werden. Die Reset-Leitung (im Schaltplan mit CLR bezeichnet) wird auf den Steuereingang MR des Bausteins gelegt. Durch diese Beschaltung ist der Eingang nur für eine kurze Zeit auf low und sonst immer auf high. Der Eingang Pin 11 wirkt als Chip-Select-Eingang, über ihn wird das IC ausgewählt. Mit der steigenden Flanke am Takteingang wird das an den D-Eingängen anstehende 8-Bit-Datenwort übernommen und steht an den Ausgängen zur Verfügung.
Der Baustein 74LS374 ist ein 8-fach-flankengetriggertes D-Flipflop mit Tristate-Ausgängen. Da die Ausgänge des Bausteins mit dem internen Datenbus der Karte verbunden sind, würde ein Baustein ohne Tristate-Ausgänge ständig einen Buskonflikt verursachen. Falls eine Busleitung auf low wäre und der jeweilige Ausgang wäre gleichzeitig auf high, käme dies einem Kurzschluß gleich.

Wie man aus der Logiktabelle (*Tabelle 3*) ersehen kann, übernimmt auch dieser Baustein seine Eingangsdaten mit der steigenden Flanke am Takteingang (CP-Eingang). Allerdings werden die Daten mit der Flanke nicht erst eingelesen, sondern aus den internen Registern auf die Ausgänge übergeben. Das bedeutet, daß die internen Register immer die aktuell anliegenden Daten speichern.

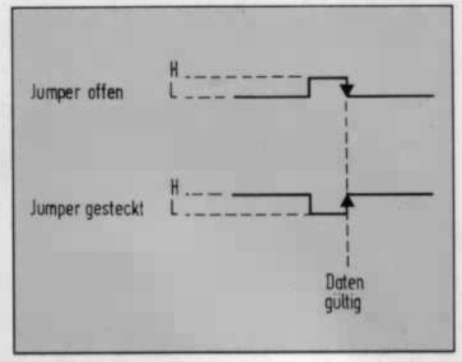


Bild 3. Signalübergänge

Die Steckbrücken

Die Basisadresse wird über Steckbrücken eingestellt (*Tabelle 4*). Auf die Details, die

Bild 2. Der Schaltplan

Tabelle 3: Die Logiktabelle des Bausteins 74LS374

	Eingänge –OE	Eingänge CP	Dn	Interne Register	Qn
Laden und Register lesen	0	^	l	L	L
	0	^	h	H	H
Lade Register Tristate	1	^	l	L	(Z)
	1	^	h	H	(Z)

(Z) High impedance (Tristate)

bei der Wahl der Basisadresse wichtig sind, wurde weiter oben schon hingewiesen. Auf der Platine befinden sich außer zum Einstellen der Basisadresse noch weitere Steckbrücken (*Tabelle 5*). Unterhalb von IC14 befinden sich die Steckbrücken J3 bis J6. Mit diesen kann das Übernahmesignal zur Datenausgabe gewählt werden. Ist die entsprechende Brücke gesteckt, werden die Daten bei der steigenden Flanke übernommen, sonst bei der fallenden (Bild 3).

Ebenso befinden sich zwei Steckbrücken im Einleseteil der Platine, um auch hier eine Option zu wählen. Beim Einlesen kann man zwischen gemeinsamem und getrenntem Strobe an den Steckern ST2 und ST1 wählen. Ist A-B gesteckt, so ist der Strobe getrennt, ist B-C gesteckt, so ist der Strobe gemeinsam (*Tabelle 6...7*). Hierbei sind die Strobe-Signale jeweils auf die Stiftleisten (Pin 9) nach außen geführt.

Aus der *Tabelle 8* geht die Belegung der Stiftleiste hervor.

Einfacher Aufbau

Der Aufbau der Karte ist denkbar einfach, da nur wenige Bauteile verwendet wurden und die Bestückungsdichte relativ gering ist. Ein besonderes Augenmerk sollte lediglich auf die Bestückung der Karte mit dem Tantal-Elko C17 und den IC-Fassungen gerichtet werden. Der Kondensaotor ist am Körper mit einem ±gekennzeichnet. Diese Kennzeichnung ist auch auf der Platine vorhanden. Die Fassungen sollten so eingelötet

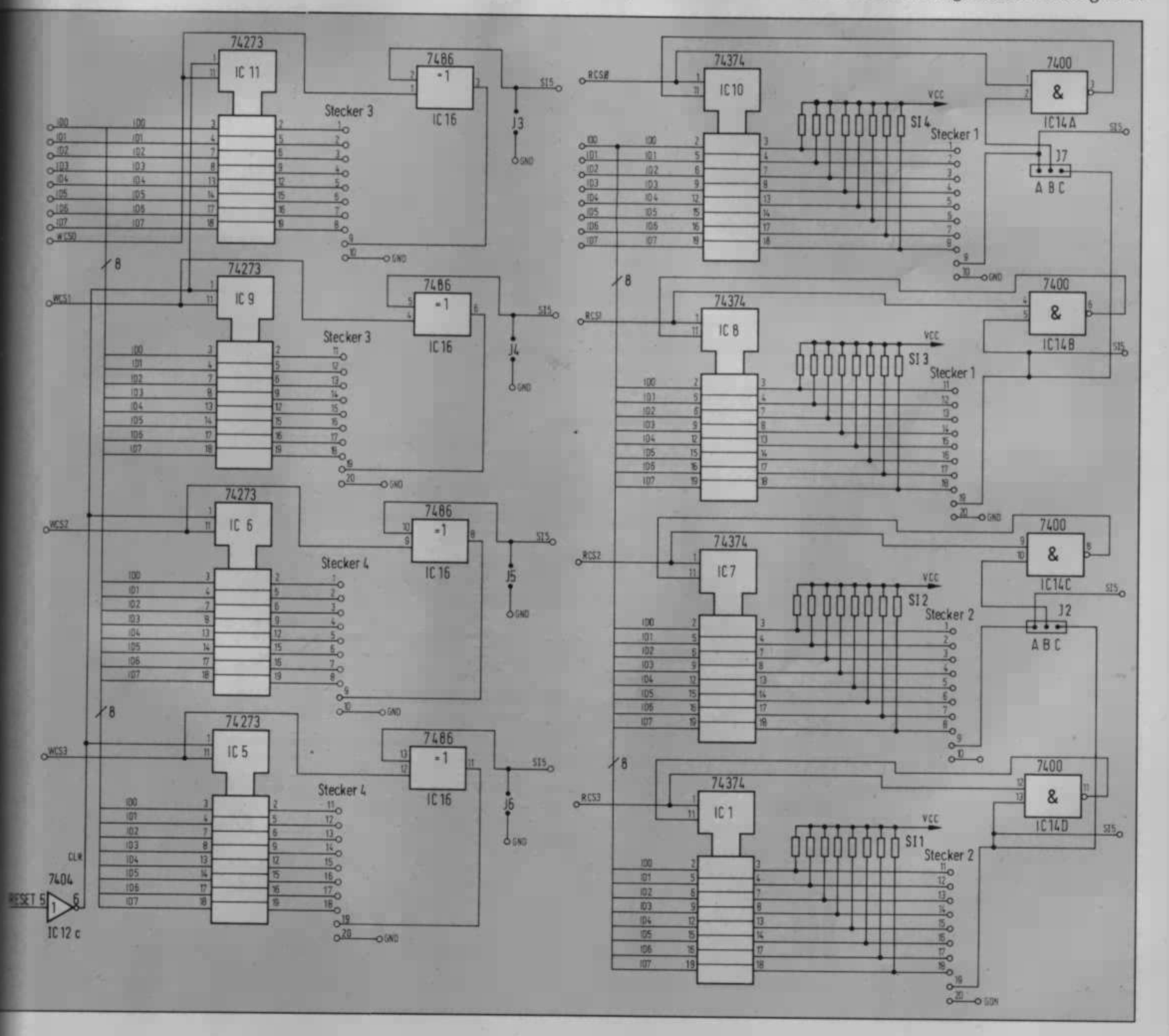


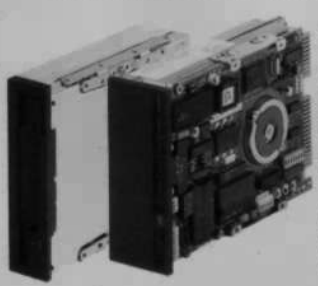
Bild 4. Der Bestückungsplan

Bild 5. Die Lötseite

ORGATECHNIK
KÖLN '88
Halle 3.1, Gang N,
Stand 63, und bei NEC
Halle 2.1/E 42, G 41

Wenn Ihr altes Laufwerk ausgeleiert hat.

NEC Laufwerke der neuen 3 1/2"-Generation sind Synonyme für dauerhafte Zuverlässigkeit. Geringes Gewicht, kleinste Abmessungen, leiser Lauf und größte Speicherkapazität sind ihre Merkmale.
NEC Laufwerke sind die idealen Massenspeicher der Zukunft. Moderne Systeme und leistungsfähige Laptop-Computer sind schon heute ohne sie nicht mehr denkbar.



Floppy Disk Laufwerk FD 1137 H

1,44 MByte (high density format)
720 KByte (normal density format)
IBM AT-Kompatibel
Media Erkennung
25,4 mm hoch, 134 mm tief
Gewicht: 400 g

Festplattenlaufwerk D3146h

51,24 MByte (unformatiert)
40,30 MByte (formatiert)
mittlere Positionierzeit unter 35 ms
automatische Transportsicherung
41 mm hoch, 146 mm tief
Gewicht: 1500 g

Wesentlich für den Erfolg aller NEC Laufwerke ist die NEC Philosophie: Erst nach ausgiebigen und härtesten Praxistests werden Neuentwicklungen den Anwendern vorgestellt. Das Ergebnis: Höchste Zuverlässigkeit und größte Datensicherheit.
Wir bei REIN liefern Ihnen sämtliche NEC Floppy Disk- und Festplattenlaufwerke von 3 1/2" über 5 1/4", 8" bis zum 9"-Format mit Speicherkapazitäten bis 1,13 Gigabyte. Rufen Sie uns an, bevor Ihr altes Laufwerk ausgeleiert hat.

REIN Elektronik GmbH, 4054 Nettetal 1
Telefon (0 21 53) 7 33-0
Fax (0 21 53) 73 31 10
Telex 8 54 203
in Bayern:
Telefon (0 89) 9 03 00 61
Fax (0 89) 9 04 35 44
Telex 5 23 317



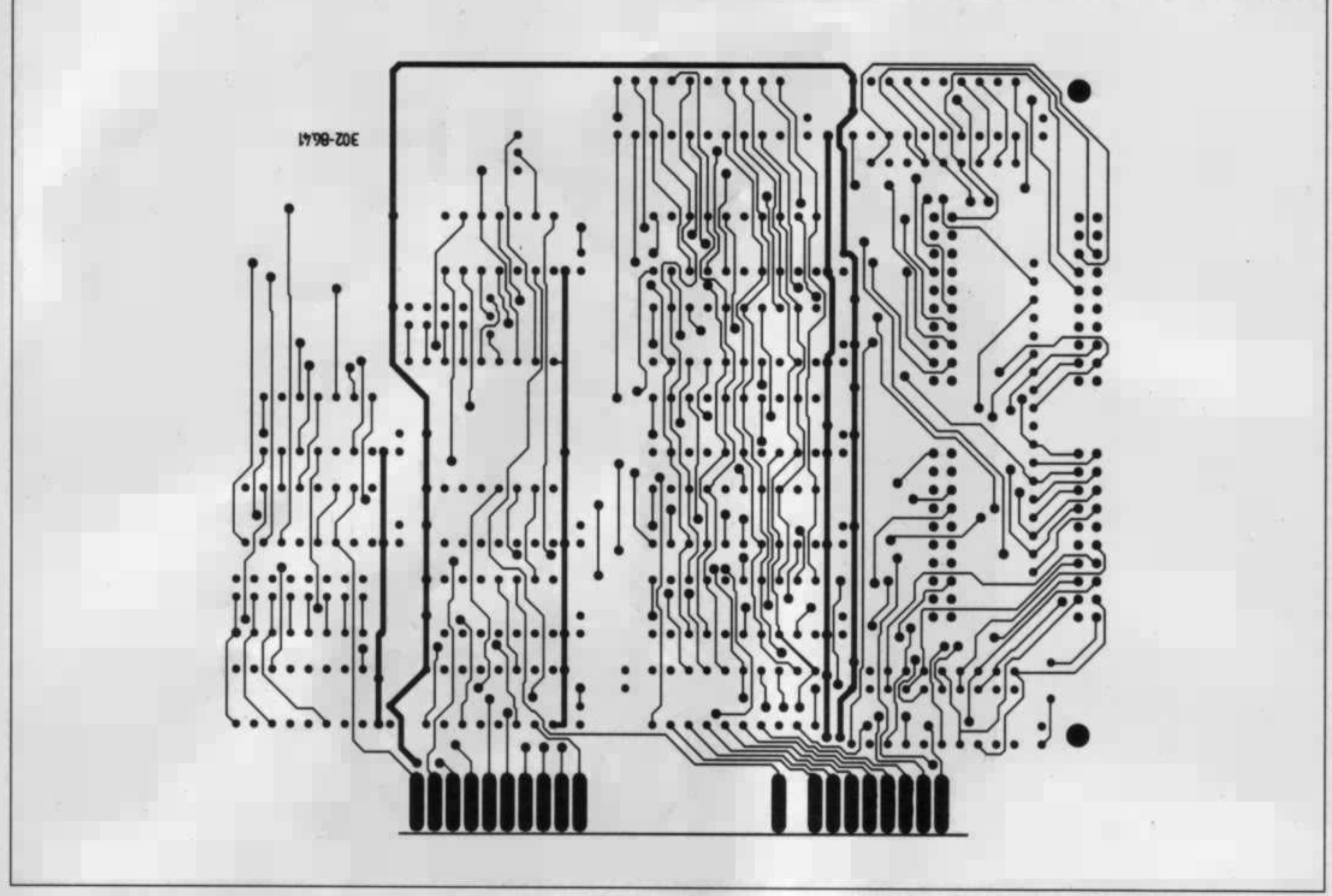


Bild 6. Die Bestückungsseite

werden, daß die Kerbe der Fassung die Position von Pin 1 der Schaltkreise kennzeichnet. Es ist sinnvoll auf der Karte für alle Schaltkreise Fassungen einzusetzen. Sollte man mal versehentlich an die Ausgänge oder an die Eingänge eine zu hohe Spannung angelegt haben, so ist es recht einfach die defekten Bauteile auszutauschen. Die *Bilder 4...6* geben alle Informationen zum Bestücken der Platine. Die Karte ist als Leerkarte und als Bausatz beim Elektronikladen Detmold, erhältlich.

Tabelle 4: Einstellbeispiele für die Basisadresse

Port-Adresse	A A9	B A8	C A7	D A6	E A5	F A4	G A3	H A2
300H	X	X	–	–	–	–	–	–
304H	X	X	–	–	–	–	–	X
308H	X	X	–	–	–	–	X	–
30CH	X	X	–	–	–	–	X	X

X: offen
–: geschlossen

Tabelle 5: So sind die Steckbrücken den einzelnen Schaltkreisen zugeordnet

Steckbrücke für	IC
3	11
4	9
5	6
6	5

Tips zur Inbetriebnahme

Sollte die Karte trotz sorgfältigen Aufbaus nicht funktionieren, so können Sie die Fehlersuche mit einer optischen Kontrolle beginnen. Kontrollieren Sie zuerst die Stellung der Steckbrücken. Dies ist besonders wichtig für die Adreßdecodierung. Bei falscher Adresse wird die Karte natürlich auch nicht angesprochen. Danach achten Sie auf kalte oder ausgelassene Lötstellen. Sollte die Suche dennoch erfolglos verlaufen, so kann man mit einem Osziloskop und einigen kleinen Test-Prozeduren den Fehler eingekreisen. Beginnen Sie mit dem Test der Adreßdecodierung. Ist die Adresse mit der Steckbrücke J1 richtig eingestellt, so müssen an Pin 2 von IC12A Nadelimpulse sichtbar sein, wenn die eingestellte Adresse angesprochen wird. Einige Zeilen eines Hilfsprogramm sind in *Bild 7* dargestellt.

Diese Impulse müssen sowohl bei einem Lese- als auch bei einem Schreibzugriff zu sehen sein. Sollte dieser Teil der Karte einwandfrei funktionieren, so kann eigentlich nur noch ein Fehler in der Read-/Write-Selektion sein. Dazu werden die Basisadressen und die darauf folgenden 3 Adresse angesprochen und an den entsprechenden

Tabelle 6: So werden die Steckbrücken für den Strobe gesteckt

	getrennter Strobe	gemeinsamer Strobe
J2	A–B	B–C
J2	A–B	B–C

Tabelle 7: Die Adreßbelegung der Stecker

Ausgabe:		
Stecker 3	Pin 1...10	Basisadresse
Stecker 3	Pin 11...20	Basisadresse+1
Stecker 4	Pin 1...10	Basisadresse+2
Stecker 4	Pin 11...20	Basisadresse+3
Einlesen:		
Stecker 1	Pin 1...10	Basisadresse
Stecker 1	Pin 11...20	Basisadresse+1
Stecker 2	Pin 1...10	Basisadresse+2
Stecker 2	Pin 11...20	Basisadresse+3

```
type str8 = string[8];
     str3 = string[3];
     line = string[255];
var  ausgabe        : str8;


procedure get_binaer(text : line ; var output : str8);
(* Diese Procedure gibt einen beliebigen Text in Zeile 25 aus
   und liefert eine binaer Byte als string zurueck              *)
var x_alt, y_alt : byte;
begin
  X_alt := WhereX;
  Y_alt := WhereY;
  GotoXY(1,25);
  ClrEol;
  write(text,'█        █');
  GotoXY(2+length(text), 25);
  Buflen := 8;
  read(output);
  GotoXY(X_alt, Y_alt);
end;

procedure dezimal_zu_binaer(input : real; var out : str8);
(* Diese Prozedure wandelt den dezimal wert input in einen
   binaer String um                                              *)
var i      :  byte;
    wert   : real;
begin
  out := '';
  for i:= 7 downto 0 do
    begin
      wert := exp(i*ln(2));
      if (input - wert ) >= 0 then begin
                                     input := input - wert;
                                     out := out + '1';
                                   end
                              else out := out + '0';
    end;
end;

procedure hex_zu_dezimal(input : str8; var output : integer);
(* Diese Procedure wandelt den Hex - Wert input in einen
   Integer - Wert um                                             *)
var i       : byte;
    ziffer : integer;
begin
  for i:=1 to length(input) do input[i] := upcase(input[i]);
  output := 0;
  for i:=1 to length(input) do begin
          Ziffer := ord(copy(input,i,1)) - 48;
          if Ziffer > 9 then Ziffer := Ziffer - 7;
          output := output + round(Ziffer * exp(ln(16) * abs(i-length(input))));
          end;
end;


procedure port_zu_binaer(adresse : integer; var output : str8);
(* Diese Procedure liest einen binaer Wert von dem Port input
   ( hex Angabe, z.B $306 ) und gibt den binaer - Wert als
   String an das Programm zurueck                                *)
var i           : byte;
    input       : byte;
    wert        : integer;
begin
  input := port[Adresse];
  output := '';
  for i:= 7 downto 0 do
```

```
    begin
      wert := round(exp(i*ln(2)));
      if (input - wert ) >= 0 then begin
                                     input := input - wert;
                                     output := output + '1';
                                   end
                              else output := output + '0';
    end;
end;

procedure binaer_zu_dezimal(input : str8; var output : integer);
(* Diese Procedure wandelt den binaer Wert input in den
   dezimal Wert output um                                        *)
var i   : byte;
begin
  output := 0;
  for i:=1 to 8 do begin
                 if copy(input,i,1) <> '0' then
                     output := output + round(exp(ln(2) * abs(i-8)));
               end;
end;

procedure binaer_zu_port(adresse : integer; input : str8);
(* Diese Prozedure gibt den binaer Wert input auf einem
   Port[Adresse] aus ( Angabe Adresse in hex : z.B  $306    *)
var dezi    : integer;
begin
  binaer_zu_dezimal(input,dezi);
  port[Adresse] := dezi;
end;

procedure binaer_and(bin1, bin2 : str8; var bin3 : str8);
(* Diese Procedure verknuepft zwei binaer Variblen und gibt die
   And - Verknuepfung binaer zurueck                             *)
var dez1, dez2 : integer;
begin
  binaer_zu_dezimal(bin1, dez1);
  binaer_zu_dezimal(bin2, dez2);
  dez1 := dez1 and dez2;
  dezimal_zu_binaer(dez1, bin3);
end;

procedure dont_care_and(Adresse : integer; bin : str8);
(* Diese Procedure vergleicht den Wert eines Port's ( Adresse in
   hex ) mit einem vorgegebenen Binaer - Wert , nicht relevante
   Bit's  werden mit einem 'x' gekennzeichnet , z.B
   dont_care_and($306, 'xxx101xx');                              *)
var  bin_p, bin_p_ver, bin_ver : str8;
     sp                        : array[1..8] of boolean;
     i                         : byte;
begin
  bin_ver := '';
  for i:=1 to 8 do begin
                 if upcase(bin[i]) = 'X' then sp[i] := false
                                          else begin
                                                 bin_ver := bin_ver +
copy(bin,i,1);
                                                 sp[i]  := true;
                                               end;
                 end;
  repeat
    bin_p_ver := '';
    dezimal_zu_binaer(port[Adresse], bin_p);
    for i:=1 to 8 do if sp[i] = true then
                 bin_p_ver := bin_p_ver + copy(bin_p,i,1);
  until bin_ver = bin_p_ver;
end;

Untertitel :  Listing 1 der Datei IO32.INC
```

Bild 10. Pascal-Prozeduren für mc-E/A-Karte

Tabelle 8: Belegung der Stiftleiste

A0	1	o	o	2	A1
A2	3	o	o	4	A3
A4	5	o	o	6	A5
A6	7	o	o	8	A7
Strobe	9	o	o	10	GND
B0	11	o	o	12	B1
B2	13	o	o	14	B3
B4	15	o	o	16	B5
B6	17	o	o	18	B7
Strobe	19	o	o	20	GND

Tabelle 9: Stückliste

IC 1, 7, 8, 10	74LS374
IC 2	74LS245
IC 3, 4	74LS85
IC 5, 6, 9, 11	74LS273
IC 12	74LS04
IC 13	74LS32
IC 14	74LS00
IC 16	74LS86
IC 15	74LS139
C1...16	10-nF-Keramik-Blockkondensator
SI 1...6	Widerstandsnetzwerk 8×3,3 kΩ
Stiftleisten 4× 2×10polig 2× 1×3polig 1× 2×4polig 1× 2×8polig	Stiftleiste, zweireihig
14	Steckbrücken (Jumper)
Fassungen 4× 14polig 2× 16polig 10× 20polig	

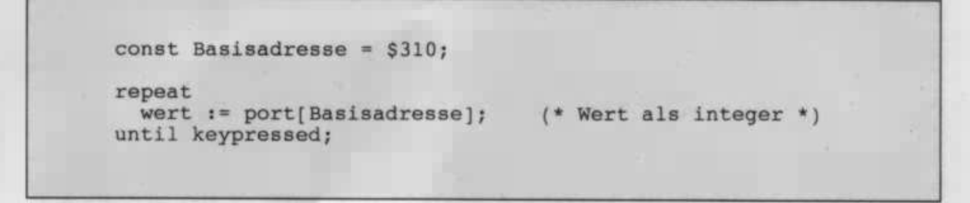
Pins gemessen. Für die Write-Selektion sind die Schaltkreise IC 5, 6, 9 und 11 zuständig. An den Pins 11 der ICs (Chip Select) werden wieder Nadelimpulse gemessen. Für den Read-Teil sind dement-

```
const Basisadresse = $310;

repeat
  port[Basisadresse] := 0;
until keypressed;
```

Bild 7. Schleife für Schreibzugriffe

sprechend die Schaltkreise IC 1, 7, 8 und 10 zuständig. Einige Zeilen aus dem Testprogramm sind in *Bild 8* gezeigt. Sollte kein Oszilloskop vorhanden sein, so können an die Ausgänge der Schaltkreise auch Leuchtdioden für den Ausgabetest angeschlossen werden (Vorwiderstand nicht vergessen).
Bei den Lese-Zugriffen werden die Nadelimpulse am Pin 1 (Output Enable) des jeweiligen Bausteins gemessen. Zusätzlich muß man diese Schaltkreisen über Pin 11



```
const Basisadresse = $310;

repeat
  wert := port[Basisadresse];     (* Wert als integer *)
until keypressed;
```

Bild 8. Schleife für Lesezugriffe

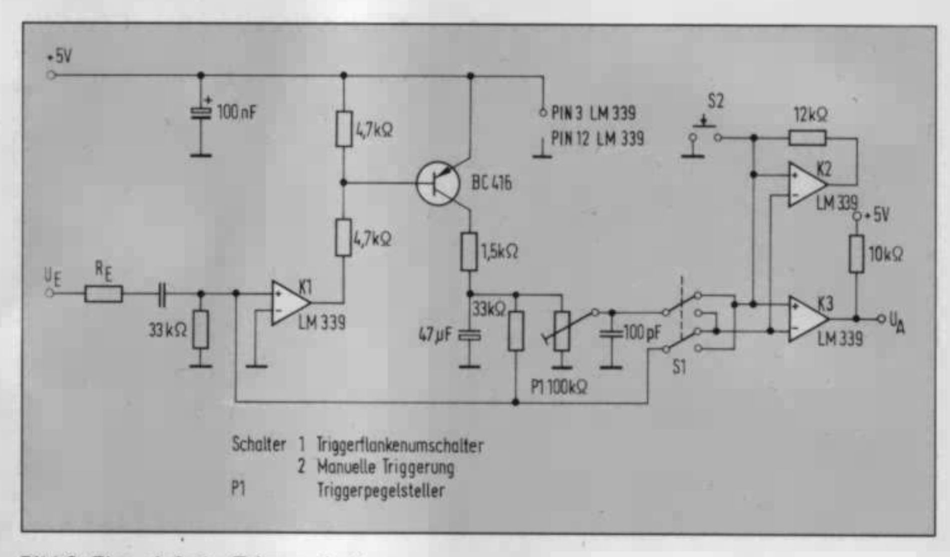


Bild 9. Eine einfache Triggerschaltung

in den Tristate-Zustand schalten, was man ebenso messen kann.
Wenn die Karte einwandfrei funktioniert, sind der späteren Verwendung keine Grenzen gesetzt. Mit ein wenig Phantasie findet man für sie viele Anwendungen. Es sei auf eine Triggerschaltung für den A/D-Wandler AD12 (siehe mc 3/88) hingewiesen, mit der man zu einem definierten Zeitpunkt die Messung startet. Dazu benötigt man eine Schaltung (*Bild 9*) [1], über die man den Triggerpegel für die Messung einstellen kann. Das entscheidende bei der Triggerschaltung ist eigentlich nur, daß sie ein beliebiges Eingangssignal so in TTL-Pegel umformt, daß es mit der E/A-Karte verarbeitet werden kann. Zur weiteren Verarbeitung der Meßwerte im Rechner ist die Prozedur dont_care_and gedacht (*Bild 10*). Ebenso kann man mit dieser Prozedur die verschiedenen Eingänge einer Alarmanlage abfragen.

Software in Turbo-Pascal

In Bild 10 sind mehrere Prozeduren für den Einsatz mit der IO32/32-Karte abgedruckt. Wenn es sich in den Prozeduren um Binär-Werte handelt, so sind diese als String definiert, bei direkter Angabe darf man nie die Hochkomma vergessen. Die Adreßangaben für die Ports sind als sedezimale (hexadezimale) Integer-Werte definiert, so daß man der Zahl in Turbo-Pascal-Manier ein Dollarzeichen voranstellen muß. Einige Prozeduren dienen lediglich zur Umrechnung zwischen den verschiedenen Zahlensystemen und sind allgemein verwendbar. Die Angaben in Binär-Werten für die Port-Ein- und Ausgabe sind sinnvoll, da man daraus leichter den Zustand der einzelnen Ports erkennen kann, als wenn man die jeweilige Dezimalzahl interpretieren muß. In der Prozedur dont_care_and können nicht relevante Bits in der Binär-Angabe mit einem X gekennzeichnet werden (z. B. xxx101xx, somit sind für die Bedingung nur Bit 2 bis 4 relevant). Am einfachsten probiert man mit der fertigen Karte und den Prozeduren deren Leistungsfähigkeit für den jeweiligen Anwendungsfall aus.

In der nächsten Ausgabe zeigen wir Ihnen, wie Sie mit dieser Karte und etwas Zusatz-Hardware aus Ihrem PC auch einen Steuerungsrechner für professionelle Anwendungen machen können.

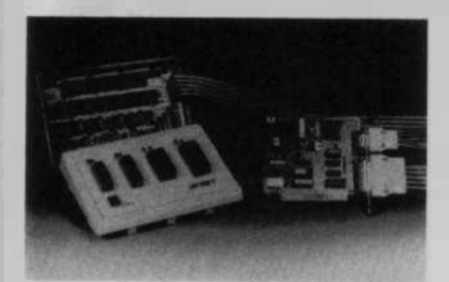
Literatur

[1] Digital- und Logikschaltungen. Elektronik Sonderheft Nr. 232. Franzis-Verlag, München.

mc-QUICKIE

mc-quickies sind aktuelle Produktanzeigen, mit denen Firmen ihre Produkte vorstellen. Verantwortlich für den Inhalt sind die Inserenten.

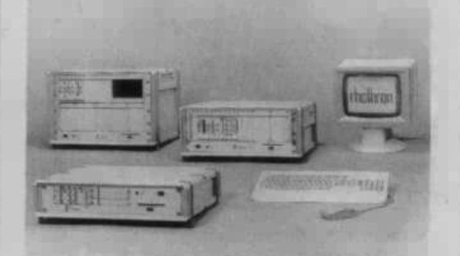
Die 5 Minuten Lösung



Von der Idee zum Chip
UP 40 Universalprogrammer
für PALs, GALs, PEELs, EPROMs, EPLDs
Preis unter 3000.–, neu PLD-Assembler
CMTE – FAST DATEN SYSTEME GmbH
Kapuzinerstr. 11, 8000 München 2
Tel. (0 89) 53 07 37, FAX: (0 89) 53 76 48

Zeichnungen ***Neu***
einfach mit dem Computer und der professionellen
BOARDmaster
E-CAD Design-Software erstellen!
Vom Schaltplan zum Layout. DEBUG-Funktion erkennt Schaltungsfehler. Netzlister generierung aus dem Schaltplan, Autorouter, Stückliste, Bestückungsplan, Bohrplan, Maske. Leistungsmerkmale: Mehrlagentechnik, SMD-fähig, Symbolmakrobibliotheken, differenzierte Gummibandtechnik, COPY, MOVE, Netzoptimierung und und ...
Auflösung bis 1/1000 Zoll auch für den **Autorouter**.
Ausgaben: EGA-Grafik 640×350 bis 800×600 oder hochauflösender Grafikkarte bis 1500×1024 Pixel, Plotter, Photoplotter, Laserdrucker, Matrixdrucker, Diskette, CNC-Steuerungen und andere. Lauffähig auf IBM PC XT/AT oder Kompatible, je nach Ausstattung des Programmpakets ab DM 1990.–
BOARDassistent zur Layout- u. Schaltplanerstellung gibt es bereits für **DM 448.–**
mit Autorouter DM 910.– DEMO-Diskette DM 39.–
Film- und Leiterplatten-SERVICE vom CAD-File des Programms schnell und preiswert.
Grafikkarten
Höchstauflösend mit bis zu 1500 × 1024 Punkten bzw. bis 65536 Farben. BildRAM bis 4096 × 4096 Punkten. Für IBM-Slot oder ECB-Bus-Systeme inkl. Software ab **DM 990.–**
Info anfordern bei
Klaus Gorny · Computer Soft & Hardware
Bülowstraße 16b · 1000 Berlin 30 · Tel. (0 30) 2 16 91 29

VME-Bus + ATARI ST



Die anwenderfreundlichen Systeme für
Meßdatenerfassung
Prozeßkontrolle
Automatisierung
Hoher Bedienerkomfort, einfache Programmierung und günstige Preise ermöglichen auch die Lösung Ihres speziellen Problems.
Fragen Sie uns!
rhothron GmbH
Rudolfstr. 14 · 7500 Karlsruhe · Tel. 07 21/6 03 11

Das kostenlose Christiani Kursprogramm

Lernen Sie heute, was morgen unentbehrlich ist.
Fordern Sie gleich das kostenlose Kursprogramm an, das Sie über unsere Lehrgänge informiert – u. a. auch über:

☐ Elektronik-Labor
☐ IC-Labor
☐ Digital-Computer-Labor
☐ Oszilloskop-Labor
☐ Radio-Fernsehtechnik
☐ Elektronisches Messen
☐ Mikroprozessortechnik
☐ Peripherie-Bausteine
☐ Kompakt-Kurs BASIC
☐ BASIC + Mikrocomputerpraxis
☐ PC-Anwendungspraxis
☐ Automatisierung
☐ Optoelektronik-Labor
☐ El. Steuerungstechnik
☐ Videotechnik
☐ Kompakt-Kurs Elektronik
☐ CNC-Technik
☐ C64 CNC-Simulator
☐ Bautechnik
☐ EDV-Grundlagen

Gewünschtes ankreuzen; Anzeige ausschneiden und einsenden an
Dr.-Ing. P. Christiani · Technisches Lehrinstitut
Postf. 35 69230 · Tel. 0 75 31/58 01 15 · 7750 Konstanz

ALLES UNTER EINEM DACH

CAD-SCHALTUNGSENTFLECHTUNG
LEITERPLATTENHERSTELLUNG EINSEITIG / DOPPELSEITIG, DURCHKONTAKTIERT, SIEBDRUCK / HEISSLUFT-VERZINNUNG
BESTÜCKUNGS-/LÖTARBEITEN, SMD-BESTÜCKUNG
MONTAGE / MATERIALBESCHAFFUNG, GEHÄUSEBEARBEITUNG
MUSTER-, KLEIN- und MITTEL-SERIEN
KURZE LIEFERZEITEN – GÜNSTIGE PREISE
EILSERVICE
ACHIM MEDINGER ELECTRONIC
KÖNIGSWINTERER STR. 116
5300 BONN 3
TELEFON 02 28/46 56 39
BÜROZEIT 10.00 – 15.00 UHR

Meßkarten für XT/AT/386

48-Kanal-I/O, 3×16-Bit-Zähler	152.–
168-Kanal-I/O, 3×16-Bit-Zähler	225.–
336-Kanal-I/O, 3×16-Bit-Zähler	339.–
64-Kanal-A/D, 24-Kanal-I/O	324.–
16-Kanal-12-Bit-A/D, 1×12-Bit-D/A	247.–
16-Kanal-14-Bit-A/D, 1×14-Bit-D/A	406.–
16-Kanal-12-Bit-A/D, Dual Slope	539.–
16 Output über Relais	454.–
32 Output über Relais	715.–
16 Optokoppler/16 Relais	427.–
Lochraster AT mit Decoder	112.–
Wire-Wrapping-Karte XT/AT/AT386	ab 62.–
8-Kanal-8-Bit-D/A	552.–
8-Kanal-12-Bit-D/A	892.–

Alle Meßkarten mit deutscher Beschreibung und Programmbeispiel in Basic und Pascal

IEEE-Karte mit Basic (Pascal.C opt.)	ab 480.–
Logic-Analyzer bis 100 MHz	ab 1980.–
Epromprogrammierer bis 1 MBit	ab 290.–
Zusatzt für Prozessorren 8051/52	136.–
Universalprogrammierer	1139.–
2 • RS488-Schnittstelle XT/AT	214.–
RS232 – RS422 Schnittstellenwandler	117.–
RS232 Repeater	97.–

Wir haben 40 verschiedene Interfacekarten auf Lager!
Fordern Sie unseren kostenlosen Katalog an.

Analog-Digitaltechnik Hans Schnellhammer
Hochhaus 3, 8091 Rechtmehring, Tel. 0 80 72/14 20
Vertragshändler: Niedermeier Computer Products GmbH
Allmannsberg 1, 8094 Edling, Tel. 0 80 39/12 95

NCSS 5 Statistik & Grafik

Ein über Funktionstasten menügesteuertes Statistikpaket aus USA, ohne komplizierte Kommandosprache. Einfach zu bedienen, flexibel im Einsatz, leicht mit anderer Software kombinierbar. Umfangreiche deutsche und englische Dokumentation. Versionen für Rechner mit und ohne Coprozessor verfügbar.

Basispaket NCSS 5.01 mit 19 Prozeduren:
- Datenerfassung im Spreadsheet, Lesen, Schreiben von ASCII-Files.
- 40 Funktionen für Datentransformationen wie exp, log, mod, random, ...
- Deskriptive Statistik wie etwa Kreuztabellen oder Histogramme, ...
- Multiple Regression, Faktoren-, Cluster-, Diskriminanzanalyse, T-Test, Varianzanalysen, GLM, Nonparametrics u. v. m. **DM 299,–**

Grafikerweiterung NCSS 5.1 Graphics mit:
- 2- und 3dimensionale Plots + Regressionsgerade, -ebene, Box-Plots, Gesichter, Profile, Stem-Plots, Andrews-Sine **DM 199,–**

Erweiterungen NCSS 5.3 & NCSS 5.4 mit:
- 5.3 Power Pack, nichtlineare Regression, logistische Regression, erweiterte Varianzanalysen, Funktionen Plotten **DM 99,–**
- 5.4 Exp. Design & Quality Control enthält 6 Prozeduren zur Versuchsplanung und 4 verschiedene Controlcharts **DM 99,–**

Komplettpaket (NCSS 5 + deutsches Handbuch) **DM 609,–**

UNISOFT
Dipl.-Volkswirt Heike Bausch, Friedrich-Ebert-Straße 15, 8900 Augsburg 22, Telefon (08 21) 9 49 59

IBM-XT-Kompatible	ab DM 998.–
IBM-AT-Kompatible	ab DM 1998.–
Apple-Kompatible + Zubehör	ab DM 99.–

und natürlich alle Erweiterungen wie:

Genius-Mouse	ab DM 118.–
RAM-Karten für XT und AT bis 12 MHz, bestückt 2 MB	ab DM 998.–
Harddisk, 20 MB	ab DM 498.–
Floppy, 2× 40 Track	ab DM 198.–
TTL-Monitor, 14"-Flatscreen	ab DM 398.–
Matrix-Drucker	ab DM 498.–
Modem	ab DM 398.–
Btx-Karte IBM	ab DM 598.–

Und Service in der eigenen Werkstatt!

Fordern Sie für einzelne Produktgruppen unsere Preisliste an. Händleranfragen erwünscht.

ASC-ELEKTRONIK-VERTRIEBS-GMBH
HIRSCHGRABEN 9–11 · 5100 AACHEN
TEL. 02 41/2 52 26

EINFÜHRUNGSPREISE!

NETZWERK-KARTEN!
100 % NOVELL-/SMC.KOMP.!
GETESTET unter NOVELL-Advance-Netware!

- ARCNET, 100 % komp., misch- u. tauschbar m. SMCs, 2 K on board, 8 K Boot-ROM opt., Plus: LED Diag., Stat. ID außen zugängl., STERN/BUS/Tw. Pair (PC110/210/250 kompatibel) = DM 331,–, 410,–, 392,–.
- ACTIV-HUB, 4 Port int. (Slot) = 352,–, ext. m. Netz. f. 4/8 P. = 572,–, 788,–
- ETHERNET, Novell NE1000 komp., 8-K-Multi-Packet-Puff., 8K Boot opt., Anschl. für Thin/Thick = 637,–
- Zubehör BNC T-/I-Conn/Term./Koax-Kabel RG 58/62 u. Tw. Pair sowie AdNet 286/ELS.
- Mengenrabatte ab 4 St., Versand p. N.N., Rückgaberecht 8 Tage, Garantie 1 Jahr. Händler-Preisliste geg. Nachweis.

CCE – Lutterrothstr. 70 – 2000 HH 20
Tel. 0 40/40 07 90, Fax 0 40/4 91 02 44

mc-QUICKIE

mc-quickies sind aktuelle Produktanzeigen, mit denen Firmen ihre Produkte vorstellen. Verantwortlich für den Inhalt sind die Inserenten.

300/1200/2400 Baud-Modem V.21/V.22/V.22bis

Tischgerät 698,–
PC-Steckkarte 648,–
Akustikcups 198,–

Modemkarte PC 1200 298,–
Kurze PC-Karte, V21/22, 300/1200 Bd vollduplex,
KS 1200, extern, sonst wie PC 1200 ...... **368,–**
Pocketmodem 1200 .... **398,–**
Alle Modems Hayes-komp. und **ohne FTZ-Nr. Anschluß an das Postnetz nicht erlaubt.**

AD/DA-Wandler 269,–
Auflösung 12 Bit, 16 AD-, 1 DA-Kanal

VIA 8255 179,–
I/O-Karte, 48 Leitungen, 16 LED, Timer/Counter

bsb
Hohenzollernring 74, 5000 Köln 1
Telefon **02 21-13 14 41**
Datentechnik GmbH





EPROM - Brenner
für 2716 - 27512/513

Neue Software V3.0
Auto-Setup & Window-Technik

Das professionelle Eprom-Programmiergerät für Entwicklung, Service, Produktion und Schulung für den Profi und engagierten Amateur. Anschluß an die Centronics-Buchse (hat jeder Laptop) keine Steckkarte im Rechner erforderlich. Das Gerät wurde in mc 1/87 vorgestellt. Aufgrund des exzellenten Preis- Leistungsverhältnis befinden sich nunmehr weit über 1500 Exemplare im täglichen Einsatz in Industrie, Entwicklung, Forschunslabors, Service und Schulung.

Software V3.0: Für PC/XT/AT und Kompatible. Auto-Setup zur automatischen Anpassung des Programmiersystems an Ihre vorhandene Hardware (Grafikadapter, Druckerschnittstelle, Taktfrequenz). Software in Window-Technik. Einfache und komfortable Auswahl der EPROM-Typen schließt Fehlbedienung aus. Standard und intelligente 1ms Algorithmen von 15 Herstellern und insgesamt ca. 150 verschiedene Eproms. 128K Daten können gleichzeitig im Speicher bearbeitet werden d.h. 27512/513 sind in einem Durchgang editier-, les- und programmierbar. Integrierter komfortabler "Fullscreen-Editor" für ASCII u. HEX. 16-Bit breite Programmierung (Normal-Lower-Higher Byte). Datenformate: INTEL,Motorla,Tectronix.

- Platine mp V2.01, Handbuch und Software mp V2.11 — DM 89,00
- Bausatz mp V2.01, Handbuch und Software mp V2.11(ohne Gehäuse) — DM 249,00
- Programmiergerät mp V2.0 incl. Software mp V3.0 und Handbuch — DM 649,00
- Software mp V2.11 incl. Quellcode für PC/XT/AT und Kompatible — DM 39,00
- Software mp V3.0 — DM 99,00
- Softline-Gehäuse (unbearbeitet wie Bild Größe:190x140x47/32 mm) — DM 29,00
- Druckerkabel für XT/AT und kompatible — DM 29,00

Versandkosten: Ausland DM 12,00 Inland DM 8,00 Versand per Nachnahme

B & P
Beate Seng Ludwig-Dürr-Str.10 7320 Göppingen Tel. 07161-75245

Soft Mail AG
vormals Ecosoft Economy Software AG
Postfach 30, 7701 Büsingen, Tel. 077 34 - 27 42

'Prüf vor Kauf'-Software

- **Großes Angebot von 'Prüf vor Kauf' - Software und Frei-Programmen: Über 4'000 Disketten für IBM-PC/Kompatible, Macintosh, Amiga, Atari ST, C64/128, Apple II. Viele deutsche Programme für Geschäft, Beruf, Privat, Schule.**
- **Software gratis, Sie bezahlen nur eine Vermittlungsgebühr von DM 14.40 oder weniger je Diskette. Wenn Sie Anwenderunterstützung vom Autoren wünschen, bezahlen Sie ihm eine geringe Registrierungsgebühr.**

Programm-Verzeichnis gratis

Bitte Computermodell angeben: Gegen Einsendung dieses Inserates erhalten Sie die 534

Diskette des Monats gratis







IBM-PC/XT/AT und kompatible

ADD-ON-CARDS

4-fach Eprom Brenner DM 568.-
16 bis 512k, 3 verschiedene Brenn-Algorithmen (z.B. Quick)
Umsetz-Software von INTEL und MOTOROLA-S - alles incl.



incl. Software+Steckkarte+Kabel+Gehäuse

Ingenieurbüro Lutger Ahlers, Mozartstr.23
8052 Moosburg, Tel. 08761/4245

IEC-BOX

Universal-Interface für IEC-Bus-Meßplätze

- Die preisgünstige Erweiterung für alle IEC-Systeme als Front-End für Meßdatenerfassung, Automatisierung und Meßtechnik
- Durch Einschubkarten flexibel erweiterbar
- Mikroprozessorgesteuert (6502/65802)
- Anschließbar auch über V.24 oder Current-Loop
- Ansprechbar aus allen Programmiersprachen

Grundgerät .... DM 1495,–
+ ges. MwSt.



ees GmbH, Pelkovenstraße 51
8000 München 50
Telefon 0 89/1 41 10 77

6502/65816

Europakarten-Rechner-Familie

- Vom Single-Board-Computer bis zum vollständigen Steuerrechner mit kompletter Peripheriekarten-Palette (digital, analog, Grafik, Bildverarbeitung, IEC, V.24, Prommer)
- 16 MB adressierbarer Speicher
- Festplatte bis 80 MB
- Programmiersprachen:
  - BASIC
  - PASCAL
  - FORTH
  - ASSEMBLER



ees GmbH, Pelkovenstraße 51
8000 München 50
Telefon 0 89/1 41 10 77

mc-MINIMARKT

PLATINENLAYOUT-PROGRAMM f. IBM PC, interaktiv, Ausgabe auf Drucker (9- und 24-Nadel) oder Plotter. **DM 98.–**, Demo DM 5.–.
Klaus Kroesen, Kastanienweg 2, 4290 Bocholt, ☏ 0 28 71/3 73 75

Deutsche Anleitung f. PC-TOOLS deluxe DM 19.80.
☏ 0 52 41/5 67 47

Endlich ist es da – das Public Domain und Shareware-Handbuch, auf das Sie schon lange gewartet haben.
PD-Handbuch (90 Seiten) DM 5.– oder
PD-Handbuch mit 2 Begleitdisketten
– Disk A: Program Reviews
– Disk B: Intro Disk DM 14.80
Lieferung per NN (+ DM 6.50) oder Vorkasse (zuzügl. DM 2.– Porto) auf Postscheckkonto Mchn., Kto.-Nr. 3403 13-807, Euroscheck oder Briefmarken.
REDYSOFT Software GmbH, Postf. 12 61, 8150 Holzkirchen G

Haftetiketten, 40 Größen, Info: Böhne, Försterweg 4, 3354 Dassel G

Probleme mit Dongle oder Keykarte? Defekt! Nicht transparent! Schnittstelle überlastet und und und...! Unsere Softwarelösung für: Autocad R9 485 DM, Personal Designer V3.0 495 DM, P-CAD V3.0 685 DM, Cadkey V3.02 465 DM, CadVance V2.02 465 DM, Tango/Protel PCB Editor V3.12 285 DM, Schema2 V2.03 485 DM, MG CAD V2.01 585 DM, Realia Cobol V3.0 485 DM, Novell V2.1 390 DM, Allstar 850 DM, PSpice Circuit V3.0 435 DM, PSpice Probe V1.1 295 DM, RangerPCB V1.23 535 DM, Genesis V2.5 395 DM, Win-Chart V1.03B 415 DM, Eplan V3.1 565 DM, Intercapital 435 DM, ElCad-88 V2.4 475 DM, CADesie V4.6 515 DM, S5-Siemens 685 DM, Abacus Ausschreibung 635 DM, PCDraft V6.2 565 DM, OrcadPCB V1.01 415 DM, Scientex V6.3 415 DM ***Bitte Urheberrechte beachten*** Info anfordern! Otto Stock GmbH, Postfach 10, D-7926 Böhmenkirch, ☏ 0 73 32/50 78,
FAX 0 73 32 41 90 G

LPKF CAD Layout System-Software – für den professionellen Einsatz – mit oder ohne Computer (IBM) zum halben Preis zu verkaufen. ☏ 0 61 08/7 76 57 G

SUCHE HARDWARE

Suche alte Videorecorder, auch Spulengeräte (auch defekt).
☏ 04 91/6 19 55 (nach 18 Uhr)

Suche MIDI-Karte für XT. F. Schneider, Lübecker Str. 15, 5000 Köln 1, ☏ 02 21/12 11 57

ELTEC Tastatur PAT-09 m. Downld. u. Softw. 550.–, Eurocom CP/M-Karte m. Syst., WS, dBASE 250.–, 96k-RAM-Karte (auf 384k umbaub.) 150.–.
☏ 0 62 21/8 15 75, ab 19 Uhr

BIETE AN HARDWARE

Neuw. HX-20 + Kassette + Drucker + viel Zubehör (NP ca. 2000.– DM) für VB 800.– DM zu verkaufen.
☏ 0 89/6 90 21 02, ab 18 Uhr

NDR-Klein-Computer: Z80 und 68008. Alle Platinen, Bücher, Monitor, Netzteil, ca. 60 % Bausatzpreise. ☏ 0 53 31/4 16 61, nach 17 Uhr

---

MC68000, komplett, V.2.2. LED, 2 MB, Sprache/Uhr; sehr viel Software. Info telefonisch. Preis VS.
☏ 02 51/61 92 67

OKI MICROLINE 192 PLUS + Einzelblatteinzug VB DM 1000.–.
☏ 09 11/63 77 78

IHR MONITOR- U. GRAFIK-SPEZIALIST
■ Preissensationen:
■ **NEC Multisync II 1529.–**
■ **EIZO 8060S:**
■ 820×620 P. **1539.–**
■ **EIZO 9070S:** 1280×800 P.
■ 50 MHz; 20–50 kHz **1998.–**
■ nur solange Vorrat reicht!
■ Fordern Sie Unterlagen und Preislisten an. Wir vertreiben eine Vielzahl Monitore, Grafikkarten, Scanner, Foto-, Schneidplotter, Plotter, Digitizer usw. der bekanntesten Hersteller zu absoluten Tiefstpreisen von Lowcost bis zu ultrahigh Resolution Profigeräten. Daneben führen wir für Sie Grafik-, DTP-, OCR-, CAD-, Textverarbeitungs- u. integrierte Software aller Art f. IBM u. kompatible Amiga u. Atari, ebenfalls zu Tiefstpreisen.
■ **Kein Ladenverkauf!!!**
■ Computerversand Chr. Karmann, Telastraße 115, 8000 München 90 G

Z80-Assembler auf IBM-PC für DM 100.–. Werner Pfundstein, Hauptstraße 2, 8061 Sulzemoos, ☏ 0 81 35/12 45

BIETE AN HARDWARE

****
AT 286 Komplett-Paket

6/12 MHz, 0-Wait-State
512 K bestückt/4 MB on Board
Uhr/Kalender akkugepuffert
Herkules-komp. Karte
Ser.-/Par.-Schnittstelle
102-Key-DIN-Tastatur dt.
1,2-MB-/360-K-Laufwerk
20-MB-Festplatte
FDD/HDD-AT-Controller
Gehäuse mit Reset/Schl.Sch.
Deutsches Handbuch
1 Jahr Garantie

DM 3198.–

RL-Computer GmbH
Hunsrückstr. 25
4020 Mettmann
Tel.: 0 21 04/8 24 26 + 5 29 31
Fax: 0 21 04/2 39 28

Andere Konfigurationen, Karten, Tastaturen, Festplatten etc. auf Anfrage. Bitte Preisliste anfordern.
****

Stepper 140Ncm 1,8/0,9° Optok. Eing. 180 DM, Netzteil für 250 DM. ☎ 0 22 53/88 54

Verkaufe 5¼"-Laufwerk NEC-FD11550 (umschaltbar 1,2 MB, 1,6 MB), praktisch neuwertig. VB 350.– Fr (Neupreis 540.– Fr). ☎ CH: 01/8 56 05 56

Hallo NDR-User: 30 Karten für 30–50 % u. Bausatzpreis; JADOS 2.1 70.– DM; RAMs 62256 à 18.– DM; 41256-06 à 19.– DM; Bücher u. Hefte 30 % u. Neupreis. ☎ 0 61 06/7 29 78, bis 23 Uhr

EC-Bus Leerprint 8opto-IN/8Rel-OUT m. W-Dog. 65.–. 37pol. Cannon Hauben 1.–. ☎ 0 72 22/8 16 35

RAMs
MOTOROLA MCM-6256, 120 ns, SUPERGÜNSTIG, Tagespreis erfragen, ☎ 0 51 71/1 80 89, W. Finck, Ulmenstr. 16, 3150 Peine

NDR-68008, 320 KB, PASCAL, FLOPPY, Gehäuse, Monitor, komplett oder teilweise zu 60 %. ☎ 0 55 22/27 49

Verkaufe SIEMENS PCD-2T; 10 Mon. alt; inkl. MS-DOS 3.2, GW-Basic 3.2, MS-Windows; 40-MB-Festpl., Farbmonitor, Tastatur u. Literatur. Preis VS. Lang Robert, ☎ 0 81 46/6 07, ab 17 Uhr

MC-68000-2, Monitor, IBM-Geh., Tast., Netzteil, 2 MB RAM, Preis 1800.–. ☎ 0 43 49/13 97

2. Hand mit Garantie
CBM 8296-D – Anfrage
CBM 8032, 8296 – Anfrage
CBM 8050, 8250 – Anfrage
CBM 3040, 4040 – Anfrage
CBM 4033, 3022 – Anfrage
CBM Drucker u. Festplatte
CBM Interfaces, CBM Kabel
Commodore Reparaturdienst
BÜCOTECH
Hauptstraße 167, 5020 Frechen
☎ 0 22 34/1 56 92

Apple II: TZ-Quattro CAD-Programm und neue Hardware: PC Transporter, (IIGS) RAM-Karten Z80H, spez. //c und GS Karten u. a. von AE, Cirtech u. v. m.! Info unter ☎ 0 30/7 45 00 06

EPROMSIMULATOR für 2716 bis 27512. Fertigplatine: **DM 278.–**, Komplettgerät: **DM 428.–**. Info bei: **HüWiCo**, Am Siepen 17, 4630 Bochum, ☎ 02 34/36 12 06

ECB-ELAB-KARTEN Grafik GDC2.0 1024 × 1024 NEC m. Softw. 400 DM; 2× RAM DM256K à 400 DM; IF1.2 2× PIO + Uhr 200 DM; HKM-CPU 120 DM; Orig.-Softw. T-Pascal + Toolbox; LISP80; 50 % u. NP. ☎ 0 40/8 31 64 30

- **Turbo XT, 10 MHz, V20, neu.**
- 1×360 kB, 101 Tast., 14"-SW-
- Mon., 20-MB-Festpl. AT-Geh.
- evtl. m. Drucker. ☎ 08238/4571

Tintenstrahldrucker Siemens PT88, IBM-Zeichensatz, Centronics, VB 900 DM; Epson-Drucker RX80/FT + Schnittst. RS232 + Centronics VB 600 DM. ☎ 0 60 31/6 25 41, ab 18 Uhr

Commodore PC-10/3 mit Garantie und Epson LX-800 Drucker (auch einzeln) zu verkaufen. ☎ 09 21/4 45 20, ab 18 Uhr

mc-CP/M-Computer mit Monitor, 2 LW, SW u. Dokum. DM 300.–. ☎ 05 51/2 40 36

Apple II+, 2 Disk, 64 KByte, zus. 10er Tast., Pascal, Handbuch, Schaltplan (alles Orig.) + A/D-Conv., I/O-6821, Literatur, Software, Monitor gn, DM 1650.–. ☎ 08 21/6 33 31

Intel Above Board, PC mit 1 MByte RAM, 1000.– DM. Intel 8087, 5 MHz, 100.– DM. ☎ 0 23 31/33 26 41

TALLY-2000-Drucker, 400 Z/s, wenige Betriebsstunden, ser. Schnittstelle, betriebsbereit, gegen Gebot. ☎ 0 63 83/72 54

VME-Bus-System, ELTEC-Eurocom3 mit VEX, 68010, 2 MB RAM, 2×5"-Floppy, 40-MB-Harddisk, Streamer-Anschluß 8RS232-Ports, OS-9 und CP/M68k, viel Softw. (auch einzeln), NP 25 000 DM, VHB 15 000 DM (kompl.). ☎ 06 31/2 05 29 27, 9–17 Uhr

mc-MINIMARKT

RAMs lieferbar: 256-12 DM 25.–; Mutterplatine XT, 10 MHz, DM 195.–; Laufwerk 1,44 MB DM 287.–; passende Controller-Kabel. Händleranfragen erwünscht.
☎ 0 63 03/43 87 [G]

DIN-A3-Plotter
Kein Spielzeug! **Bausatz** kompl. m. Gehäuse u. Interface nur **DM 298!** Fertiggerät nur **DM 398!** Bauplan **DM 10!** Auflösung 0,1 mm, Geschwindigk. ca. 70 mm/s. Kostenl. Info bei Peter Haase, Weissenberger Weg 226, 4040 Neuss 1,
☎ 0 21 01/53 08 12, ab 17 Uhr [G]

Magnetkartenleser.
☎ 05 71/4 98 56 [G]

---

286 Speed-Karte nur 798.– DM, 10 MHz und 8 KByte Cache-Speicher, **Ihr XT+Comp. schneller als AT!** Hdl.-Anfragen erwünscht! **FSM-TROGISCH,** Gartenstr. 10, 3051 Suthfeld,
☎ 0 57 23/8 19 46 [G]

SCHRITTMOTORSTEUERUNG
XYZ-Achsensteuerg. f. Computer mit Parallelport. Mit Netzteil und 3 Schrittmotoren **DM 269.–**
BOHRPROGR./C64 **DM 98.–**
Schrittmotor DM 29.–. Info DM 2. **PME,** Hommerich 20, 5216 Rheidt, ☎ 0 22 08/28 18 [G]

Wahnsinns-Preise
für Farbbänder, Disketten usw.
Preisliste anfordern
BüCOTECH
Hardware, Software, EDV-Zub.
Postfach 17 64-M, 5020 Frechen
☎ **0 22 34/1 56 92, Fax 5 77 73**

HARDWARE-MESSWERT-ERFASSUNG für fast alle Computer (siehe mc-Quickie).
L. Bockstaller [G]

KONTAKTE

Entwickle SW für IBM-kompatible Rechner. Dipl.-Inform. M. Pest, Lindwurmstr. 68, 8000 München 2

Suche Elektron., der MIDI-Schn.-Stelle für mich (XT) baut, oder mir bei Entwicklung hilft.
☎ 02 21/12 11 57, Schneider

Ing. entwickelt Hard- u. Software, Z80-ECB EPAC-Systeme, Schnittstellen-Interfaces... Professionelle CAD u. Elektronik-Werkstatt vorh.
☎ 00 43/27 49/3 24 82

68000, 6809, Dipl.-Ing. entwickelt Hard- und Software, EPAC-Systeme, intel. Interfaces, IEC-Bus.
☎ 0 69/39 38 84

Wir suchen Menschen, die Programmierer werden wollen. Durch Fernkurs zu fundierten Kenntnissen als BASIC-, PASCAL- oder COBOL-Programmierer. Ohne besondere Vorbildung werden Sie Computer bedienen, Programmsprachen beherrschen, Programme entwickeln, testen und anpassen können. Als zukunftsorientierte berufliche Weiterbildung oder interessante Freizeitbeschäftigung. 52 weitere Fernkurse. Kostenlosen Studienführer und Probelektion anfordern. Kein Vertreterbesuch. **Studiengemeinschaft Darmstadt, Abt. 36/02, Postfach 41 41, 6100 Darmstadt,** ☎ **0 61 57/80 66** [G]

HX-20-CLUB und PC-PROFI-GRUPPE c/o Jürgen und Susanne Miersch, Hindenburgd. 58a, 1000 Berlin 45

VERSCHIEDENES

Haben Sie Probleme mit Dongle oder Keykarte? Berichten Sie uns kurz schriftlich, welcher Art diese Probleme sind. Wir arbeiten an Softwaremäßigen Lösungen. Otto Stock GmbH, Postfach 10, 7926 Böhmenkirch, Fax 0 73 32-41 90 [G]

Suche MC- u. CT-Ausg. von Anfang bis 1985, Lit. ü. CP/M sowie Softw. für CP/M. ☎ 0 72 34/66 81

---

* PAL/GAL-Programmierservice *
Kein Ladenverkauf! Info anfordern von Andreas Haack, Heilmannring 59a, 1000 Berlin 13 [G]

Eine eigene Firma
muß kein Wunschtraum bleiben. Neuartige Wirtschaftszeitschrift liefert X Beispiele, wie Sie erfolgreich Ihr eigener Chef werden. Gratisinfo von: Die Geschäftsidee, Th.-Heuss-Str. 4/MC835, 5300 Bonn 2

Verkaufe mc 3/82–87; Funkschau 72–86; Chip 8/83–87, nur kompl. DM 200.–. ☎ 0 22 75/43 89

Layouter übernimmt nebenberuflich Entflechtung von Leiterplatten auf CAD-Anlage.
☎ 0 61 50/5 37 32, ab 15 Uhr

Verkaufe mc 4/84–12/85, DM 80.
☎ 02 08/37 09 69, 18–21 Uhr

---

profi*MIX von profi*pool e.V. und QuickBASIC 4, QuickC, C 5, Pascal 4, MASM von Microsoft. Sparen Sie Programmiermonate bei Umstell. auf Mixed Language. Rat u. Info (kein Verkauf) von L. Uhrmann, Mercatorstr. 10, 4250 Bottrop 2, ☎ 0 20 45/8 31 33

Software-Entwicklung in Assembler für die Prozessoren Z80/68000 und 8048/49/51. D. Kriesell, Badgasse 6, 8531 Lonnerstadt

---

Verdienst m. eigenem Computer. 24seit. inter. Info, 21.30 DM (davon 1.30 Porto). G. Hein, In der Worth 12, 3155 Edemissen

---

EDV-Literatur
Liste kostenlos: H. Weidinger, Postf. 21 05 46, 8500 Nürnberg 21 [G]

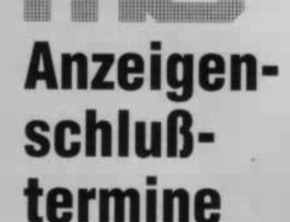
Leiterplatten: Sofortdienst für Muster, Layouts, Entwicklungen.
☎ 0 77 02/22 88 u. 18 84 [G]

---

* BESTÜCKUNG * MONTAGE * PRÜFUNG * Layout- und Leiterplattenservice * Vom Muster bis zur Serienproduktion. Jede Leiterplatte maschinengelötet, sichtkontrolliert und blitzblank ultraschallgereinigt. Ein schnelles Preisangebot erhalten Sie nach Einsendung von Bestückungsplan und Stückliste oder Bestückungsmuster. gk-electronics, Hofrichterstr. 34, 5000 Köln 80, ☎ 02 21/63 98 97, Fax: 02 21/63 42 37. Produktion: Adamstr. 37,
☎ 62 50 85 [G]

Durchkontaktierte Leiterplatten; Oberfläche Pb/Sn ungeschmolzen, Klein- und Mittelserien, Muster in wenigen Tagen, auch einseitig. **Bestückungen – Layouts.** M. Ernst, Elektron. Systeme, Postfach 8 41, 5800 Hagen 1,
☎ 0 23 31/40 30 07 [G]

STAUBSCHUTZHAUBEN –
EIN PROBLEM WENIGER –
z. B. für IBM-PC und Kompatible, zweiteilig (Zentraleinheit mit Monitor und Tastatur).
NUR DM 40.– zzgl. Versandk.
Gesamtpreisl. DM 3.– Briefm.
K * S-VERSAND
K. Schellhammer, Dorfstr. 26, 2061 Kleinwesenberg,
☎ 0 45 33/35 66
Fax 0 45 33/52 87 [G]



Anzeigen-schluß-termine

Heft 12/88 vom	28. 11. 88
Anzeigenschluß	24. 10. 88
Heft 1/89 vom	27. 12. 88
Anzeigenschluß	21. 11. 88
Heft 2/89 vom	30. 1. 89
Anzeigenschluß	2. 1. 89

mc-MINIMARKT

VERSCHIEDENES

Raum Aachen/Köln: Diskettenkonvertierung CP/M nach **MSDOS** und umgekehrt für 25.– DM per Disk. **Siegfried Müller, Hauptstraße 98, 5163 Langerwehe** [G]

MAC-TOS-DOS-OS2-9: Konvertieren v. Disketten 3,5"–5¼"-HD. ☏ 02 51/86 42 61, ab 20 Uhr

** Bohren von Leiterplatten **
– Präzise und termingetreu
– Klein-, Mittel- und Großserien
Gerloff-Elektronik,
☏ 0 51 46/86 81

Das Postspiel: GALAKT.-STAR-WARS. Bestehen Sie den Überlebenskampf im Weltall. Info von Astro-Data-Club, Postfach 943, 6400 Fulda

Verdienst m. eigenem Computer 24seit. inter. Info, 21.30 DM (davon 1.30 Porto). G. Hein, In der Worth 12, 3155 Edemissen

* Sonderposten-Copy-Cards * behebt BACKUP-PROBLEME geschützter Software: Info, ☏ 08 21/51 99 98 [G]

Service Manual Amiga 2000 für DM 100.–. ☏ 0 89/91 76 59, abends

Wir haben ein 68008-System mit 4 MByte Adreßraum entwickelt. Es ist ein Slot-Computer, der universell ausbaubar ist. Praktisch jedes Betriebssystem kann angepaßt werden. Ein eigenes in Modula 2 geschriebenes Betriebssystem wurde entwickelt. Außerdem gibt es einen in Assembler geschriebenen Modula 2 Einpass Compiler. Wir suchen Firmen und Leute, die sich mit Geld und Know-how an unserer Firma beteiligen.
Außerdem vertreiben wir bereits ein 68000/68010/68020 Europakartensystem für gehobene Ansprüche, das in der Leistung besser als die IBM-PS/2-Serie ist und außerdem billiger. LB Computer, Postfach 12 53, 8425 Neustadt/Donau

Orig.-**Cannon-**Hauben, 37- und 50pol., à **DM –.75.** ☏ 0 72 22/8 16 35

FLOPPY AM 6502-EMUF und anderen Rechnern. Bauanleitung mit Schaltplan u. 6502-Routinen geg. DM 20.– VK od. NN. 3,5"-Diskette + DM 10.–. M. Meyer, Adamstr. 48, 1000 Berlin 20

Eigener Computer bringt Geld, 24seitige interessante Info, 20 DM + 1.40 Porto. G. Hein, In der Worth 12, 3155 Edemissen

Auftrag für Gelegenheitsanzeige in mc

Anschrift: Franzis-Verlag GmbH, Anzeigenabteilung, Karlstraße 41, 8000 München 2

Name und Adresse ______

Unterschrift/Ich zahle sofort nach Rechnungserhalt. Datum

☐ Preis für private Gelegenheitsanzeige je Druckzeile 4.– DM. Chiffregebühr je Anzeige 8.– DM.

☐ Preis für gewerbliche Gelegenheitsanzeige je Druckzeile 14.82 DM (dürfen nicht unter Chiffre erscheinen). Bitte beachten Sie, daß **Preise in Anzeigen die MwSt.** enthalten müssen.

mc-programmbörse
☐ suche Software
☐ biete an Software

mc-minimarkt
☐ suche Hardware
☐ biete an Hardware

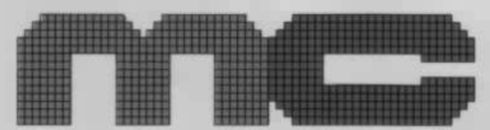
☐ Tausch ☐ Verschiedenes
☐ Kontakte ☐ Chiffre-Nummer
☐ vollständige Adresse
☐ Telefon

(Zutreffendes bitte ankreuzen)

Ich wünsche folgenden Text in der nächstmöglichen Ausgabe zu veröffentlichen:

Bitte mit Schreibmaschine oder in Druckschrift in Groß- und Kleinbuchstaben ausfüllen.

INSERENTENVERZEICHNIS



Die Mikrocomputer-Zeitschrift

REDAKTION

Anschrift:
Franzis-Verlag GmbH
Karlstraße 37–41, 8000 München 2
Postfach 37 01 20, 8000 München 37
Sekretariat: Rita Schleser
Telefon: 0 89/51 17-3 54
Telex: 5 22 301
Telefax: 0 89/51 17-3 79

Chefredakteur:
Dipl.-Math. Ulrich Rohde, verantw.
(Anschrift der Redaktion)

Stellv. Chefredakteur:
Dipl.-Ing. (FH) Ulrich Kruppe

Redaktion: Horst Brand, Dipl.-Ing. (FH) Wolfgang Hascher (fl), Aktuelles: Reiner Schönrock, PC-Technik: Dipl.-Ing. (FH) Dieter Strauß

Ständige Mitarbeiter (zu erreichen unter der Anschrift der Redaktion): Helga M. Schmidt, Günther Sternberg

Korrespondent:
USA: Ray Duncan

Art Direction: Dipl.-Des. (FH) V. Hilbel

Layout, Grafik, Herstellung:
Josef Wurzinger

Labor: Neucom Electronic GmbH

Software-Service:
Shamrock Software-Vertrieb,
Telefon 0 89/59 54 68

Sonderdrucke: Jakob Wintersberger

Lizenzen: Georg Geschke, Königstr. 8, 3000 Hannover 1, Tel. 05 11/34 53 62

Gesamtherstellung: Franzis-Druck GmbH, Karlstraße 35, 8000 München 2, Telefon 0 89/51 17-1

VERLAG

Anschrift:
Franzis-Verlag GmbH
Karlstraße 37–41, 8000 München 2
Postfach 37 01 20, 8000 München 37
Telefon: 0 89/51 17-1
Telex: 5 22 301
Telefax: 0 89/51 17-3 79
Postgirokonto München 5758-807

Geschäftsführer:
Peter G. E. Mayer,
Michael-Alexander Mayer,
Dr. Harald Wiebking

Anzeigenleitung: Dietger Kötter, verantw.
(Anschrift wie Verlag)

Anzeigenverkaufsleitung: Hans-Joachim Hecht (-3 86)

Disposition: Ingrid Daschner (-2 97)

Vertriebsleitung: Herbert Barnehl

Abonnement: Christa Fischer (-2 40/-2 79)

Handelsverkauf: Dietlind Stauder (-2 04/-2 83)

Bezugspreise Inland: Einzelheft 8,– DM, Jahresabonnement 70,– DM; Vierteljahresabonnement 18,50 DM. Studenten, Auszubildende und Rentner erhalten das Jahresabonnement gegen Nachweis günstiger. Der Versand ist im Abonnementspreis eingeschlossen. In den Preisen ist die gesetzliche Mehrwertsteuer in Höhe von 7% enthalten.

Die mc erscheint monatlich, jeweils montags am Monatsanfang bzw. am Ende des Vormonats; im 8. Jahrgang.

ISSN 0720-4442
Vertriebskennzeichen B 7745 E



VERLAGSVERTRETUNGEN

Anzeigenvertretung Inland:
Baden-Württemberg: Stefan Zwickl-Dorschel, VerlagsService, Gerlingerstr. 6, 7257 Ditzingen, Tel. 0 71 56/3 39 24/25

Bayern: Elfie Rusch, Münchner Verlagsvertretung, Sperberstraße 23, 8000 München 82, Tel. 0 89/4 39 10 33

Berlin: Rainer W. Stengel, Bischofsgrüner Weg 91, 1000 Berlin 46, Tel. 0 30/7 74 45 16

Hessen: Dieter Blank, Verlagsservice, Bockenheimer Landstraße 33, 6000 Frankfurt/M. 1, Tel. 0 69/71 40 03 55

Norddeutschland: Lita Lange, impulse medien service GmbH, Holtenklinker Str. 9, 2050 Hamburg 80, Tel. 0 40/7 24 20 37/38

Nordrhein-Westfalen: IMEDIA Medien-Vertretungs GmbH i.G., Bolkerstraße 57, 4000 Düsseldorf 1, Tel. 02 11/8 00 37

Anzeigenvertretung Ausland:
Schweiz: Exportwerbung AG, Kirchgasse 50, CH-8024 Zürich, phone: 01-47 46 90, telex 8 12 765

Großbritannien: Martin Geerke, 4, Friary Hall (Flat 3), Friary Road, South Ascot, Berks SL5 9HD, U. K., phone: 9 90-2 86 49, telex: 858 328 EUROAD

Japan: ABC Enterprises Inc., Heinz W. Kuhlmann, 7-4, Ohyama-cho, Shibuya-ku, Tokyo 151 Japan, Tel. 4 85-29 61-3, Fax 4 66-07 09

USA: International Media Marketing, 7310 Adams St. Suite C, Paramount, CA 90723, Tel. (2 13) 4 08-19 69, Fax: (2 13) 6 33-11 55 (Balboa Supply).

France: Agence Gustav Elm, 41, Avenue Montaigne, 75008 Paris, phone: 01-47 23 32 67

Italia: Rancati advertising, Milano San Felice Torre 5, I-20090 Segrate, phone: 02-7 53 14 45, telex: 3 11 250 PPMII

Belgien: ECI/United Media Int. S.A., Avenue de la folie chanson, 2 bte 7, 1050 Bruxelles, Tel. 02/6 47 31 90, Telex: 6 3 950 eci um

Anzeigenpreise nach Preisliste Nr. 8, gültig ab 1. 10. 1987

Verlagsvertretungen Ausland (Bezugspreise in Klammern):

Belgien: Office International des Périodiques (O.I.P.), Avenue Marnix 30, B-1050 Brüssel (Einzelheft 200,– bfr, Jahresabonnement 1963,– bfr)

Dänemark: Harck + Gjellerups Booksellers Ltd., Fiolstraede 31–33, DK-1171 Kopenhagen K. (Einzelheft 32,– dkr, Jahresabonnement 316,– dkr)

Frankreich: Librairie Parisienne de la Radio, 43, rue de Dunkerque, F-75010 Paris

Luxemburg: Messageries Paul Kraus, 5, rue de Hollerich, Luxembourg

Niederlande: De Muiderkring BV, Electronics House, Postbus 313, 1380-AH Weesp (Einzelheft 9,50 hfl, Jahresabonnement 164,50 hfl)

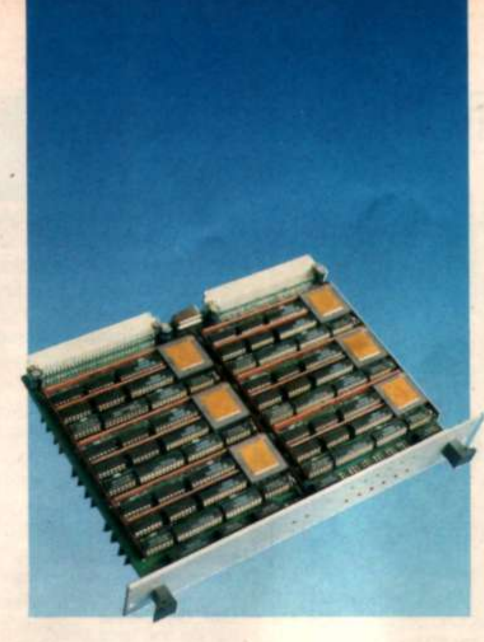
Österreich: Erb-Verlag Ges.m.b.H. & Co., KG, Buch- u. Zeitschriftenvertrieb, Amerlingstr. 1, A-1061 Wien (Einzelheft 60,– öS, Jahresabonnement 620,– öS)

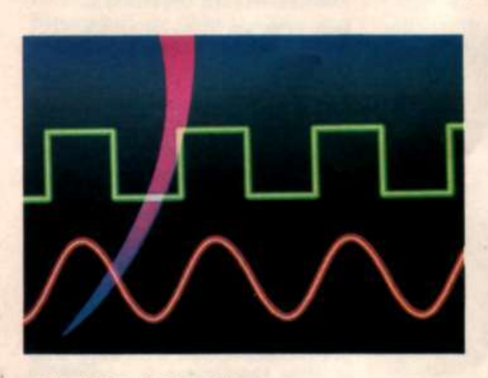
Schweiz: Verlag Thali AG, CH-6285 Hitzkirch (Luzern) (Einzelheft 8,– sfr, Jahresabonnement 67,20 sfr, je nach Kurssituation)

VORSCHAU

Transputer

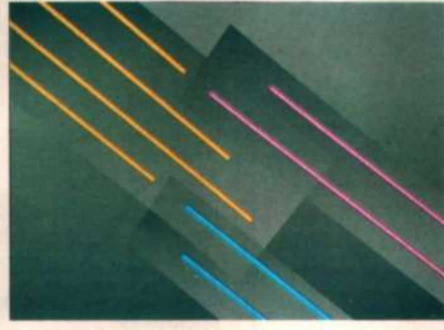
Professor Schulten und seine Mitarbeiter von der Arbeitsgruppe theoretische Biophysik im Physik-Department der TU München zeigen, wie man mit Transputern einen massiv-parallelen Rechner aufbaut, der beim Einsatz in der Erforschung von Biomolekülen die Leistung einer Cray erreichen kann – aber nur ein hundertstel davon kostet. Mit Schaltplan und Programm! Ein einzelner Rechnerknoten kostet nur ca. 2000.– DM. Mit fünfzig davon ist Prof. Schulten so schnell wie die Cray. Es ist überraschend, was man mit Transputern erreichen kann.





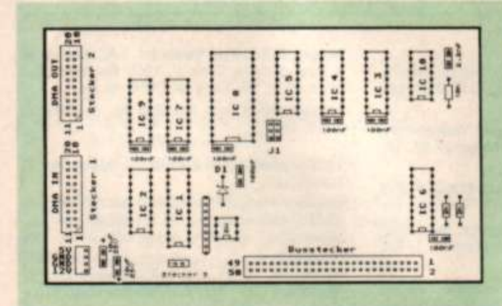
AD/DA-Wandlung

Es geht nicht nur darum, irgendwie AD/DA-Wandlungen durchführen zu können, es kommt auf die Qualität an. Ein Fachmann auf dem Gebiet schildert, wie man solche Wandler dynamisch analysieren kann.



Theorie

Lisp, der Klassiker der Programmiersprachen für KI, wird heute mehr denn je genutzt – allerdings vorwiegend für anspruchsvollste Projekte. mc zeigt, wie man ein semantisches Netz in Lisp aufbaut.



Festplatte an Atari

Mit mc können Sie Ihren Atari ST mit einem Interface ausrüsten, das den Anschluß preiswerter Karten aus dem PC-Bereich gestattet. Mit einem Omti-Controller und einer PC-Festplatte bekommt man zum Beispiel einen unschlagbar schnellen Massenspeicher.



Modula-2

Diesmal ein Spiel, das lernen kann. Tic-Tac-Toe ist zwar sehr bekannt, aber gerade deshalb können daran die programmtechnischen Dinge, wie sie der Autor realisiert hat, leicht durchschaut werden. Außerdem die Fortsetzung der Serie „Ein Betriebssystem in Modula-2". Mit dieser Reihe wollen wir allen Mut machen, die bisher die Techniken des Multi-User und Multi-Tasking-Betriebs mit Mißtrauen betrachtet haben. Der Beitrag zeigt die Stärken von Modula, der Sprache zur Konstruktion von Betriebssystemen.

Materialien zu mc

Rufen Sie 0 89/59 54 68 an, es meldet sich die Firma Shamrock, die in unserem Auftrag die Sammeldisketten für je 25 DM vertreibt: fragen Sie nach der jeweils neuesten.

MS-DOS-Sammeldisk 19: Turbo-Pascal ohne Absturz und Linien in C aus mc 1/88; GALs programmieren aus mc 1–2/88, Festplatte sicherer gemacht; Rentenberechnung; Batch-Dateien mit mehr Komfort und Bildschirm-Attribute in Turbo-Pascal aus mc 2/88.

MS-DOS-Sammeldisk 20: Pascal-Programm zur Simulation des Know-how-Computers aus mc 4/88.

MS-DOS-Sammeldisk 21: Parameterübergabe aus Turbo-Pascal aus mc 4/88, CGA-Emulation mit Hercules-Karte und schnelle SIN- und COS-Algorithmen aus mc 5/88.

MS-DOS-Sammeldisk 22: Programmieren unter OS/2, Assembler mit Struktur und VGA-Test aus mc 6/88; Modula-Programm zur Braitenberg-Welt aus mc 7/88.

MS-DOS-Sammeldisk 23: Alle mc-Jahresinhaltsverzeichnisse von 1981 bis 1988 als ASCII-Dateien, inklusive Suchprogramm.

ST-Sammeldisk 1: EPROM-Programmer und Taschenrechner aus mc 9/86 (TOS-Version vom 27. 11. 86 oder später nötig!)

ST-Sammeldisk 2: Grafikdemo und Minimon aus mc 5/88.

ST-Sammeldisk 3: 3-D-Darstellung von Funktionen in GFA-Basic aus mc 7/87, Assembler-Routinen in GFA-Basic und Programme zum Setzen des Uhrenbausteins DS1216E aus mc 9/87.

ST-Sammeldisk 4: Physik und Gehirn aus mc 9/87, Split-Speed und Co-Routinen in Modula-2 aus mc 11/87, div. Public-Domain-Programme (Achtung: zweiseitig formatierte Diskette!)



Das KI-Sonderheft von mc enthält kompaktes Wissen über das Gebiet der künstlichen Intelligenz. Lisp, Expertensysteme, KI-Programmiertechniken, Prolog, maschinelles Beweisen, Physik und Gehirn – das sind die wichtigsten Themen, die im Sonderheft zu finden sind. Es kostet 19.50 DM.

Ausgabe 11/88 erscheint am 24. Okt. 1988



Unser preiswertester Komplett-PC:
Commodore PC 1
Ein vollwertiger IBM-Kompatibler. Mit allem, was ein PC braucht:
Prozessor: Intel 8088, 4,77 MHz
Speicher: 512 Kbytes, auf 640 K erweiterbar
Diskettenlaufwerk: 5,25", 360 KBytes formatiert
Schnittstellen: Serielle RS-232-Schnittstelle, 8-Bit-Centronics-Parallelschnittstelle, ext. Diskettenschnittstelle, Schnittstellen für Erweiterung, Maus, Tastatur, RGBI-Video, Composite-Video.
Video-Modes:
Monochrom-Text: 80 x 25 Zeichen,
Hercules Grafik: 720 x 348 Punkte,
Color-Text: 40 x 25 Zeichen, 16 Farben,
80 x 25 Zeichen, 16 Farben,
Color-Grafik: 320 x 200 Punkte, 4 Farben,
640 x 200 Punkte, 2 Farben
Unterstützte Monitore: Farbmonitor (RGBI digital), Monochrom-Monitor digital oder Composite (BAS)
PC 1 (ohne Monitor)
Incl. MS DOS 3.2
+ GW BASIC
nur 699.-
Auch bei PORST
Unten:
HIGHSCREEN HD 1
FESTPLATTEN:
HD 1-30 30 MB Festplatte für PC 1 incl. Anschlußkabel. Sensationspreis nur
895.-
HIGHSCREEN COMPUTER MONITOR
HIGHSCREEN
Commodore PC-1
Commodore
Commodore PC-1
komplett nur
798.-
Auch bei PORST
1599.-
NEUE FILIALEN:
Seit 3.8. auch im Finnlandhaus, HAMBURG, Esplanade 41, Telefon 040/35 36 58
Ab Mitte September auch in 8720 SCHWEINFURT, Markt 12 Telefon 09721/18 53 13
Ab 1.10.'88 NEUE ADRESSE in BIELEFELD: Alfred-Bozi-Str. 14
WICHTIG: Herstellerbedingte Lieferzeiten! Auf Grund erhöhter Nachfrage ist nicht immer alles sofort lieferbar!
VOBIS MICROCOMPUTER
kompetent und preiswert
HAUPT-VERWALTUNG:
Postfach 1778
Rotter Bruch 32-34
5100 AACHEN
☎ 0241/50 00 81
Tx 832 389 vobis d
1000 BERLIN 30
Kurfürstenstr. 101 · 030/2 13 94 80
2000 HAMBURG
Krohnskamp 15 · 040/2 79 46 76
2300 KIEL
Sophienblatt 74-78 · 0431/67 86 22
2800 BREMEN
Violenstraße 37 · 0421/32 04 20
3000 HANNOVER
Berliner Allee 47 · 0511/81 65 71
4000 DÜSSELDORF
Wielandstr. 21 · 0211/35 99 64
4100 DUISBURG 1
4150 KREFELD
4300 ESSEN
Huyssenallee 3 · 0201/23 17 74
4600 DORTMUND
Hamburger Str. 110 · 0231/57 30 72
4800 BIELEFELD
Herforder Str. 106 · 0521/6 38 78
5000 KÖLN
Mathiasstr. 24-26 · 0221/24 86 42
5100 AACHEN
6000 FRANKFURT
Frankenallee 207/209 · 069/73 40 49
7000 STUTTGART
Marienstr. 11-13 · 0711/60 63 36
6400 FULDA
Mittelstr. 19/21 · 0661/7 82 66
(gemeinsam mit Foto PORST)
7500 KARLSRUHE
Kriegsstr. 27/29 (am BGH) · 0721/37 82 68
7750 KONSTANZ
8000 MÜNCHEN
Aberlestr. 3 · 089/77 21 10
8500 NÜRNBERG
Vordere Ledergasse 8 · 0911/23 29 95
8900 AUGSBURG
Jakoberstr. 16 · 0821/152349
Deutschlands umsatzgrößter Microcomputer-Spezialist